IMAGE UNDERSTANDING AND INFORMATION EXTRACTION

PURDUE UNIVERSITY, LAFAYETTE, INDIANA

SEPTEMBER 1976

RADC-TR-76-303
Interim Report
September 1976

IMAGE UNDERSTANDING AND INFORMATION EXTRACTION

Purdue University

Sponsored By
Defense Advanced Research Projects Agency (DoD)
ARPA Order No. 2893

ROME AIR DEVELOPMENT CENTER
AIR FORCE SYSTEMS COMMAND
GRIFFISS AIR FORCE BASE, NEW YORK 13441

The images projected in this report are actual images and thus clarity/lack of clarity, is important.

This report has been reviewed by the RADC Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

This report has been reviewed and approved for publication.

APPROVED: David J. Brazil

DAVID J. BRAZIL, Capt, USAF
Project Engineer

Do not return this copy. Retain or destroy.

ia

IMAGE UNDERSTANDING AND INFORMATION EXTRACTION

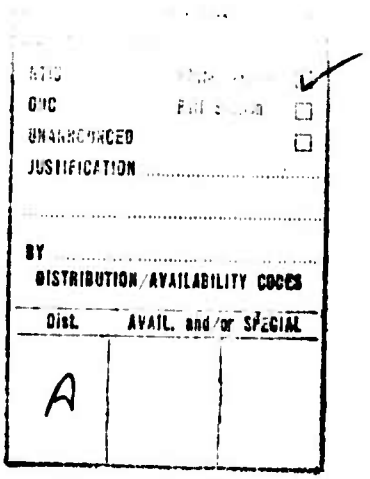
T. S. Huang
K. S. Fu

Contractor: Purdue University
Contract Number: F30602-75-C-0150
Effective Date of Contract: 1 November 1975
Contract Expiration Date: 31 October 1976
Short Title of Work: Image Understanding and Information Extraction
Program Code Number: 5D30
Period of Work Covered: Nov 75 - Jan 76

Principal Investigators: Dr. Thomas S. Huang & Dr. King Sun Fu
Phone: 317 493-3361
Project Engineer: Capt David J. Brazil
Phone: 315 330-3175

Approved for public release;
distribution unlimited.

This research was supported by the Defense Advanced Research Projects Agency of the Department of Defense and was monitored by Capt David J. Brazil (IRRE), Griffiss AFB NY 13441 under Contract F30602-75-C-0150.



ib

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER RADC-TR-76-303	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) IMAGE UNDERSTANDING AND INFORMATION EXTRACTION		5. TYPE OF REPORT & PERIOD COVERED Interim Report Nov 75 - Jan 76
		6. PERFORMING ORG. REPORT NUMBER N/A
7. AUTHOR(s) T. S. Huang K. S. Fu		8. CONTRACT OR GRANT NUMBER(s) F30602-75-C-0150
9. PERFORMING ORGANIZATION NAME AND ADDRESS Purdue University School of Electrical Engineering West Lafayette IN 47907		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61101E B8930001
11. CONTROLLING OFFICE NAME AND ADDRESS Defense Advanced Research Project Agency 1400 Wilson Blvd Arlington VA 22209		12. REPORT DATE September 1976 13. NUMBER OF PAGES 113
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Rome Air Development Center (IRRE) Griffiss AFB NY 13441		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Same

18. SUPPLEMENTARY NOTES

RADC Project Engineer: Capt David J. Brazil (IRRE)

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Digital image processing; Image segmentation; Edge detection; Image texture classification; Fourier shape descriptors; Syntactic scene analysis; Missile

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report summarizes the results of Purdue's research program on Image Understanding and Information Extraction supporred by the Defense Advanced Research Projects Agency under Contract F30602-75-C-0150. The report covers the period 1 Nov 75 to 31 Jan 76.

The objective of this research is to achieve a better understanding of image structure and to use this knowledge to develop techniques for image analysis and processing tasks, especially information extraction. The emphasis is on

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

the syntactic decomposition and recognition of imagery based on scene analysis. It is our hope that the results of this research will form the basis for the development of technology relevant to military applications of machine extraction of information from aircraft and satellite imagery.

TABLE OF CONTENTS

IMAGE UNDERSTANDING AND INFORMATION EXTRACTION

RESEARCH OVERVIEW AND SUMMARY

Since October 1, 1973, we have been supported by ARPA IPT office to carry out research in image analysis and modeling. More recently, the emphasis of our research has been shifted toward achieving a better understanding of image structure and using this knowledge to develop techniques for extracting information from images. To reflect this new emphasis, we have changed the title of our program to: Image Understanding And Information Extraction.

The ultimate goal in automatic image understanding and information extraction is to build a machine which will look at a scene and then give us a qualitative as well as quantitative description of it. This formidable task can be broken down into several sub-tasks as indicated by the block diagram in Fig. 1.

The sensor looks at the scene and outputs image data which are generally multispectral and multitemporal. The preprocessing box attempts either to put the image into a form which is more suitable for analysis by the latter boxes or to compress the image data for transmission to a remote location or storage for future use. The image segmentation box, in simple instances, locates objects in the image (for example, in character recognition, it locates the characters). For more complex scenes, it segments the image into regions. Each of these regions is classified by the classification box. The classification can be done either by classical decision-theoretic methods (feature extraction followed by statistical classification) or by the more recent syntactical methods. In linguistic terminology, the regions are primitives, and the classifier finds attributes for these primitives. Finally, the structural analysis box attempts to find the relationship (spatial, temporal, spectral) among the regions (primitives). In some cases, one may want to develop a grammar for the primitives.

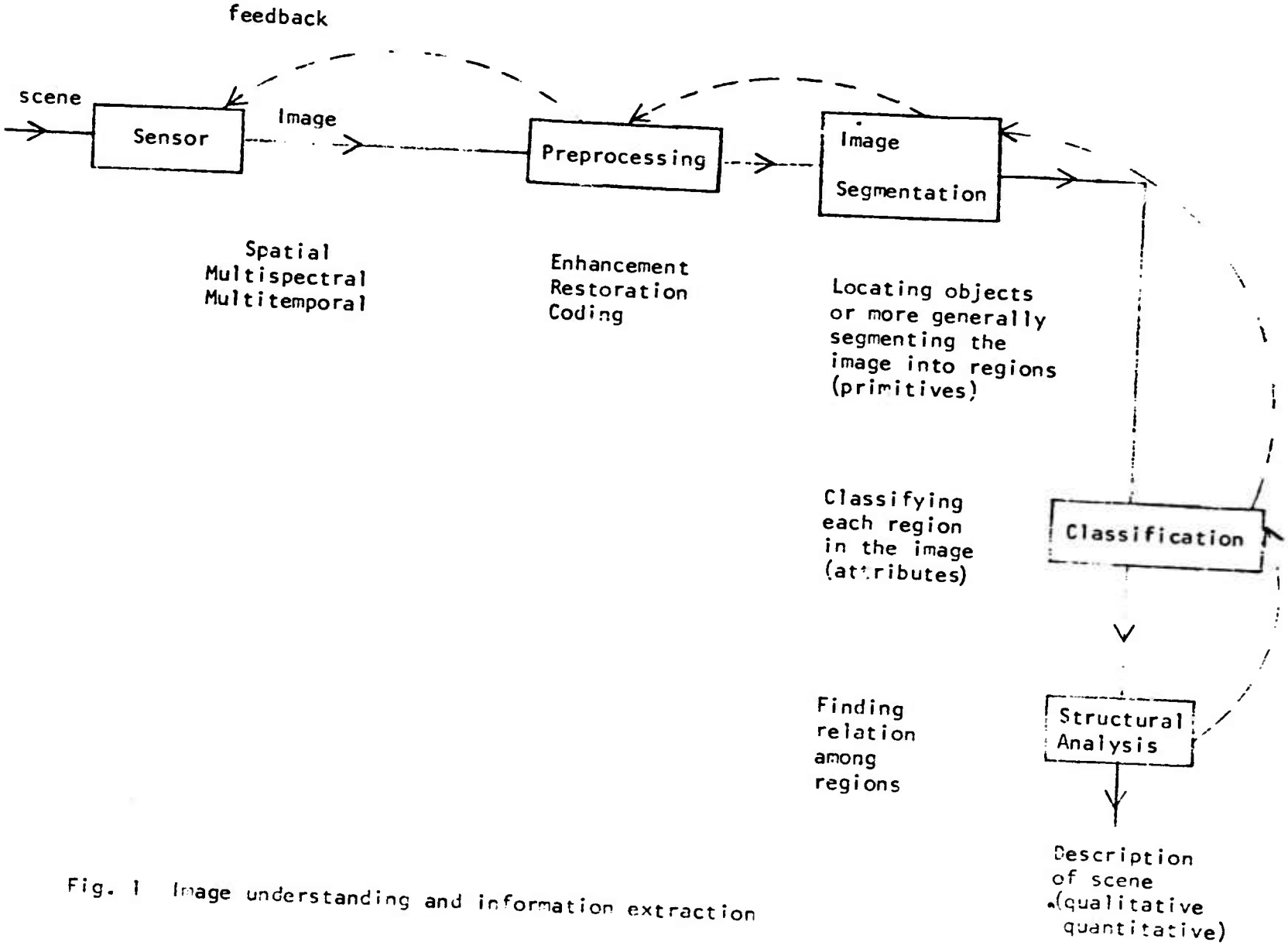


Fig. 1 Image understanding and information extraction

The functions of the boxes in Fig. 1 overlap, and their demarcation is not clear cut. Furthermore, there is strong feedback in the system (as indicated by the dashed arrows in the block diagram). For example, structural analysis may reveal that certain regions are classified wrongly and thus force a reclassification. Similarly, difficulties encountered in classification may make it necessary to reexamine image segmentation. Etc. We aim to study not only the individual boxes in Fig. 1, but also their interaction.

We are carrying out both basic and applied research with the hope that on the one hand the results of our basic research could find practical applications, and on the other hand problems arise from practical situations would guide the direction of our basic research. Our emphasis in both basic and applied research is on the syntactic decomposition and recognition of imagery. The data base we use includes but is not limited to aircraft and satellite multispectral imagery.

Our research projects fall into five overlapping categories.

A. IMAGE PRIMITIVES: Segmenting images into regions (primitives)

We investigate two approaches to image segmentation: edge detection, and region growing.

(1) Edge Detection

Many edge detection techniques have been studied. However, none of them works well for noisy and unsharp images. There are two difficulties. First, noise gives rise to false edge points. Second, noise and image blurring make it very hard to connect edge points to form continuous curves.

The main thrust of our research is to use syntactic methods to help extract connected edge curves in a noisy and unsharp environment. We propose to employ a "two-level" scheme to detect edges where there is

a-priori knowledge of some structure in a pattern. The basic idea is to separate the problem of detecting the edges in noise from that of detecting the edge structure, but to improve the efficiency of each solution by transferring information from one to another. To locate noisy or "smeared" edges, a parser for a stochastic string generating grammar will be used, but its operation will be controlled by a web grammar parser that recognizes the edge structure. This assumes a parallel processing capability so that the two levels may proceed simultaneously and guide each other. The use of stochastic web grammars would not be intended to handle "local" noise and smearing, but rather any such phenomena which have a structural character (e.g., edges obscured by shadows).

In order to use syntactic methods to help edge detection in a digital environment, it is necessary to understand the nature of digitized edges. We are therefore studying the characterization of digital edges, especially straight edges and lines. Some properties of ideal digital straight lines and a method of detecting them are discussed in the report by Tang and Huang. Currently, we are experimenting with real digital edges, i.e., edges obtained by digitizing real-life images which contain blurring and noise.

Another important problem we are studying that is closely related to edge detection is mensuration. In many reconnaissance applications, one is often interested in measuring the distance between two edges. Here, it is of the utmost importance to determine edge locations accurately. We have developed an efficient algorithm for finding the maximum a posteriori estimates of edge locations which works very well in our preliminary tests (see the report by Burnett and Huang). Currently we are testing the algorithm on aerial photos.

(ii) Region Growing Techniques

The BLOB algorithm of Wintz and Gupta works quite well. However, because of its sequential nature, the resulting boundaries depend on the order in which the picture elements are processed. For example, if we process the image in the conventional line scanning order, then we tend to get many more vertical than horizontal boundaries.

To remedy this, we propose to develop parallel segmentation techniques. One approach is to first divide the image into say 2x2 picture element regions, then we merge together regions with similar characteristics (e.g., spectral contents, mean and variance of the gray levels, or more generally, texture) in a parallel manner. All pairs of neighboring regions are examined simultaneously, and the boundary between a pair is removed if the two regions are similar. This process is repeated until all neighboring regions are dissimilar.

A more efficient initialization (i.e., to create starting regions for the parallel region growing algorithms) can be achieved by using unsupervised clustering techniques. Each picture element is represented by a point in the feature space. The features could be, for example, spectral contents of the picture element, means and variance of the spectral contents (or more generally texture) of a small neighborhood around the picture element in question. The feature space is then partitioned by some clustering algorithm. The picture elements in each cluster are considered as in the same region in the original image. The region is generally disconnected - it contains several connected subregions.

The main thrust of our current research in image segmentation is to develop efficient algorithms to do the parallel segmentation including

initialization, and to find (for various classes of images) suitable measures of similarity between the regions. Some preliminary results of unsupervised clustering using various feature pairs are contained in the report by Yoo and Huang.

After we have gained experience in parallel merging and similarity measures, we shall move on to the development of hierarchical segmentation techniques and the interaction between segmentation and structural scene analysis.

B. IMAGE ATTRIBUTES: Characterizing the primitives

(1) Texture Analysis

The success of the region-growing approach to image segmentation depends critically on the suitable choice of a similarity measure between regions which in turn is largely a problem of texture analysis. We plan a major research effort in the understanding of image textures and the use of textures in image information extraction.

One major problem of texture analysis is that there is no common definition of image texture. Our definition will be an iterative development starting with sample textures from a book by Brodatz, photographs of natural scenes, aircraft data, and LANDSAT imagery. As we develop texture descriptors and classification algorithms, our texture definitions will be modified to incorporate the new findings.

The main thrust of our current research is to find texture descriptors for pattern classification which are invariant to illumination and resolution changes, and are computationally simple; and to use these texture descriptors to find texture boundaries in images.

We have developed a new class of texture descriptors, the max-min descriptors, which are computationally very simple and which work

extremely well in our preliminary classification and boundary finding experiments. Some results of texture boundary finding are shown in the report by Mitchell.

Our study on texture is closely related to our study on image segmentation, since similarity measures among regions will no doubt include texture descriptors. Texture will probably be relied on heavily in hierarchical image segmentation.

It is evident that texture can be defined on several levels; e.g., the detailed texture of a small rock and the overall texture of hundreds of rocks. Complex images may be described in terms of several levels of textures. By adjusting the parameters in an image segmentation algorithm, we can first divide the iamge into regions according to micro-texture. Then we merge some of these regions together according to the next higher-level texture, etc. This approach is particularly suitable for a syntactic description of the image. Depending on the particular application at hand, we will consider the regions at a certain texture level as the primitives. Then a grammar will be derived to describe how these primitives combine to form higher-level regions.

(ii) Shape Analysis

We are attacking the shape analysis problem both from the classical decision-theoretic point of view and the more recent syntactic approach. In the classical framework, the first step in pattern recognition is feature extraction. And one important class of features in picture processing and recognition is that of shape descriptors. Recent work by Fu and Persoon indicates that Fourier boundary descriptors are quite effective for classifying hand-printed numerals and some medical images.

In terms of optimal curve matching in shape analysis, a distance measure has been proposed. The distance measure is expressed as the Euclidean metric between the two sets of Fourier descriptors representing two different curves. The minimization of the distance measure reduces to the solution of roots of a periodic function with a finite number of harmonics. This is also one of the advantages of using Fourier descriptors instead of performing the minimization of a distance measure defined directly in the spatial domain.

We are refining the Fourier descriptors and applying them to the shape analysis of military-related patterns. Specifically, we are using the Fourier boundary descriptors for airplane shape recognition. The results are very encouraging (see the report by Wallace and Wintz).

C. IMAGE STRUCTURE: Syntactic scene analysis - Determining the Relationships Among Primitives

Much of the work to date with respect to the computer analysis of image data (and especially multispectral image data) from high-flying aircraft and earth-orbiting satellites has concentrated on the spectral domain; i.e., the characterization of each point in the image (on the ground) in terms of its "color". Little attention has been paid to the information contained in the spatial relationships in the data, even though past experience with photo-interpretation indicates that this information component is substantial. We are studying the characterization of the spatial information content of such imagery in two ways: (1) syntactic image models, and (2) Markov dependency models of context.

Although the syntactic or "linguistic" approach to image description and analysis has been widely cited as holding great promise, little practical work of this nature has yet appeared. However, Fu and

Brayer have recently made progress in this direction, developing a grammar for the description of roads and for the detection of clouds and cloud shadows in satellite imagery. An extension of the work of Fu and Brayer is contained in the report by Fu, et al.

The power of the syntactic approach in accounting for pattern structure (both spatial and, potentially, spectral dependencies) makes it a natural avenue to pursue for the description and "understanding" of such imagery. Specific applications include the detection and classification of man-made objects; detection of geological structures indicative of ore deposits and suitable construction sites; and land-use and natural resource inventory.

Classifiers which operate in the spectral domain generally classify a single image point at a time. Utilization of context - the information contained in local spatial dependencies among image points - has been shown to improve significantly the accuracy with which classification can be performed and images described. One way to characterize such dependencies is in terms of Markov processes, i.e., statistical transition properties. We conjecture that characterization of local scene dependencies in this manner could provide useful "picture primitives" for input to syntactic models such as we have discussed above.

The objective of this aspect of the research is to determine the utility of two-dimensional (and possibly higher-dimensional) stochastic dependencies for characterizing spatial relationships and improving classification in image data. These methods will be applied to image data in two ways: as part of the classification operation per se, and in a post classification mode where the input has been "pre-classified" using its multispectral characteristics. The impact of parallel

processing on the efficiency of this type of analysis will be assessed.

More details on the use of context are contained in the report by Fu, et al.

D. IMAGE RECOGNITION TECHNIQUES.

Many pattern recognition problems may be viewed as combinatorial optimization problems. Exhaustive enumeration can determine the optimum solution, but the required computation is prohibitive. One of the promising approaches to overcome this difficulty is the branch and bound method. In exhaustive search, we search all branches of a tree into the bottom and evaluate a criterion vlaue for each path. In the branch and bound method each node of the tree provides information as to whether or not the search goes down further. If the answer is no, the search bounces back from the node without trying the branches under the node. Thus, the number of computed paths is expected to be significantly smaller.

Our objective is to apply this technique effectively to pattern recognition. Many problems in pattern recognition can be reformulated in terms of the branch and bound method. In order to meet our objective, we must find, for each problem, a proper criterion which evaluates the value of each branch and also can provide the needed information at each node. The problems we are considering include: feature selection, clustering, and classifier design.

E. APPLICATIONS

(1) Locating Large Concrete Airports in LANDSAT Imagery

We first use spectral information to locate concrete areas in the image. Then we find straight line segments in these areas. Finally, syntactic information will be used to separate airports from highways. Some results will be presented in a future report.

(ii) Missile Tracking

The main thrust is to develop real-time techniques for locating and tracking a missile in TV images. Edge detection and change detection techniques will be used. Because of the real-time requirement the processing time is very limited.

Surface acoustical wave scanners, being developed at Purdue, seem to be suitable for this application. A preliminary imaging experiment using a SAW scanner is described in the report by Gunshor and Huang.

(iii) Reducing Clouds and Haze in LANDSAT Imagery

By modeling clouds and haze as a multiplicative process, we attempt to reduce their obscuring effects by three-dimensional (2 spatial dimensions, 1 spectral dimension) digital filtering. Some initial computer simulation results are shown in the report by Mitchell and Chen. Currently, the technique is being applied to LANDSAT imagery.

IMAGE DECOMPOSITION

J. W. Burnett and T. S. Huang

In previous work [1] we have investigated some of the algorithms for image noise reduction available if we model an image as a finite state Markov process. This model has led to non-linear processing methods that are superior to current linear methods in images with sharp edges [1].

Currently we are studying the application of the model to the problem of image segmentation.

Segmentation algorithms are valuable in pattern recognition [3] and data compression [2] and potentially valuable in making measurements of objects characterized by gray level boundaries. Unfortunately the problem of image segmentation is not as simple as the application of a derivative operator to find boundaries. Edges are blurred by camera motion, lenses, atmospheric turbulence, sampling, etc. and there is always noise due to film grain, quantization, transmission, and so forth. Thus any segmentation algorithm must be capable of dealing with image blur as well as noise.

In this report we examine an algorithm based on our finite state model that produces maximum likihood sequence estimates of the boundary component of digitized images. This algorithm has the capability to deal with image blur and noise with great speed. While the technique has no immediate extension to two simultaneous dimensions we will suggest a way that a two-dimensional problem can be handled one-dimension at a time.

The Problem

Suppose that we model an image scan line $\underline{I}$ as a finite state Markov process as shown in Fig. 1. This model assumes that the gray level across a scan line is piecewise constant and takes on a finite number of possible gray level values. Actual scan lines do not look like Fig. 1, however, so we model

the available scan line $\underline{Z}$ as being the ideal line $\underline{i}$ passed through the system of Fig. 2.

A reasonable way to extract the boundary component of the available line $\underline{Z}$ is to calculate the maximum a posteriori estimate $\hat{i}$ of i. In other words we want to find an estimate $\hat{I}$ that maximizes $P(I|Z)$.

A well known solution to the problem is due to Viterbi [4] (see also [5] and [6]). The Viterbi algorithm not only provides a map estimate of the boundary component but requires computation time on the order of N where N is the number of points to be processed.

However, before the Viterbi algorithm can be used some modifications have to be made. First, the possible states that the boundary process can assume may not be known a priori. In this case the possible states must be obtained from the available data $\underline{Z}$. There are a number of ways that this can be done.

The technique used here is a clipped local average, which has the advantage of being very fast. There are many clustering techniques that might yield more accurate estimates but the increased accuracy would require more computation time.

Another problem that must be dealt with is that in picture processing the blurring function H is typically a function of two variables and a modification of the algorithm must be made. In many instances H will be separable and the image can be processed row by row and then column by column. If H is not exactly separable it can be broken up into a separable sum [7]:

$$H = \sum_i \sqrt{\lambda_i}\ \underline{K}_i\ \underline{\rho}_i$$

where the various terms are defined in [7]. Thus even though the degrading system H is two-dimensional we can process the image a line at a time.

Example

A 256x256 letter "G" was blurred with an impulse response

$$H = \underline{P}\,\underline{P}^T \qquad P = \begin{bmatrix} .01 \\ .21 \\ .56 \\ .21 \\ .01 \end{bmatrix}$$

Noise was added to produce a SNR of 20 dB (see Fig. 3). This noisy blurred image was processed by our algorithm. As shown in Fig. 4 the noise was reduced and the smearing eliminated. The computer time required was 241 seconds. However, this includes 130 seconds required to transpose the image with the only available routine. A more efficient transposition routine would greatly decrease the time required.

Fig. 5 shows a scatter plot of a scan line of the estimated image plotted about the scan line of the original image. In this line (and in fact most lines) the boundary locations were estimated correctly. Thus our algorithm should be useful for making accurate measurements of objects in digitized images.

Conclusions

The algorithm promises to provide a fast and accurate method of segmenting a noisy and blurred image into regions of constant gray level.

Some additional work will be necessary before the promise is realized. A careful examination of Fig. 4 shows that a few errors in estimation were made. Most of these are due to a poor determination of the true picture levels by our local averaging techniques (see Fig. 5 for example). Thus our next task is to find better ways of determining the possible states from the blurred and noisy data.

REFERENCES

[1] J. Burnett and T. S. Huang, "Image Decomposition," in Image Analysis and Modeling Interim Report, 1975.

[2] D. Graham, "Image Transmission by Two Dimensional Contour Encoding," Proc. IEEE, Vol. 55, No. 3, March 1967.

[3] J. Gupta and P. Wintz, "Multi Image Modeling," Purdue University Technical Report, TR-EE 74-24, September 1974.

[4] A. Viterbi, "Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm," IEEE Trans. Info. Theory, Vol. IT-13, April 1967.

[5] G. Forney, "The Viterbi Algorithm," IEEE Trans. Info. Theory, Vol. 61, No. 3, March 1973.

[6] -----, "Maximum Likelihood Sequency Estimation in the Presence of Intersymbol interference," Proc. IEEE, Vol. 61, No. 3, March 1973.

[7] S. Treitel and J. Shanks, "The Design of Multistage Separable Planar Filters," IEEE Trans. Geosci. Electronics, Vol. GE-9, No. 1, January 1971.

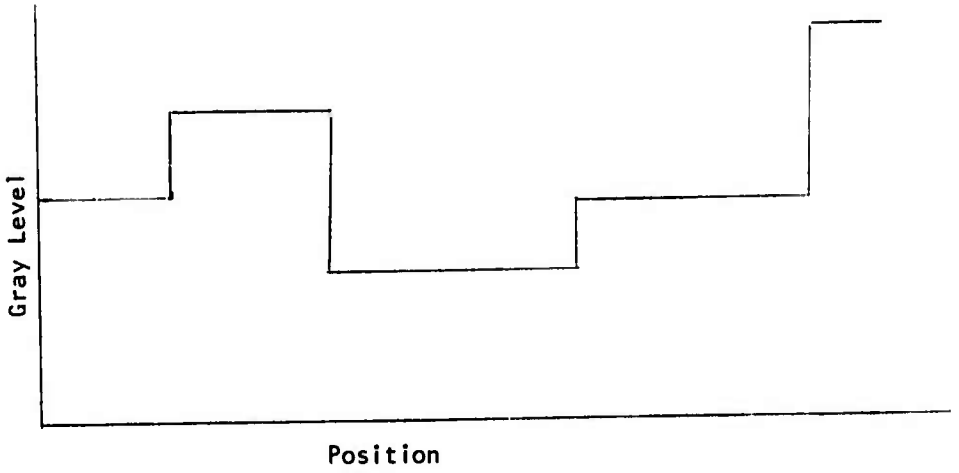
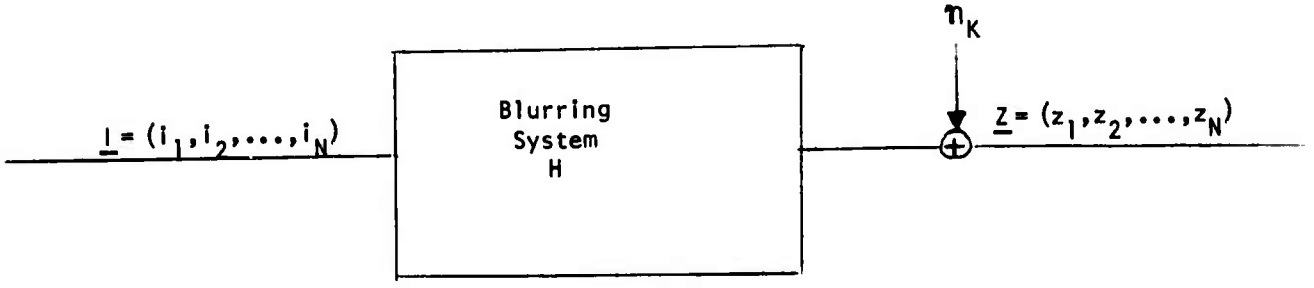


Fig. 1 A Finite State Markov Model of a Scan Line



i_K = gray level of ideal scan line at position K

z_K = gray level of actual scan line at position K

Fig. 2 Model of Scan Line Degradation

Fig. 3 The Available Image

Fig. 4 The Estimate of the Boundary Component of the Image

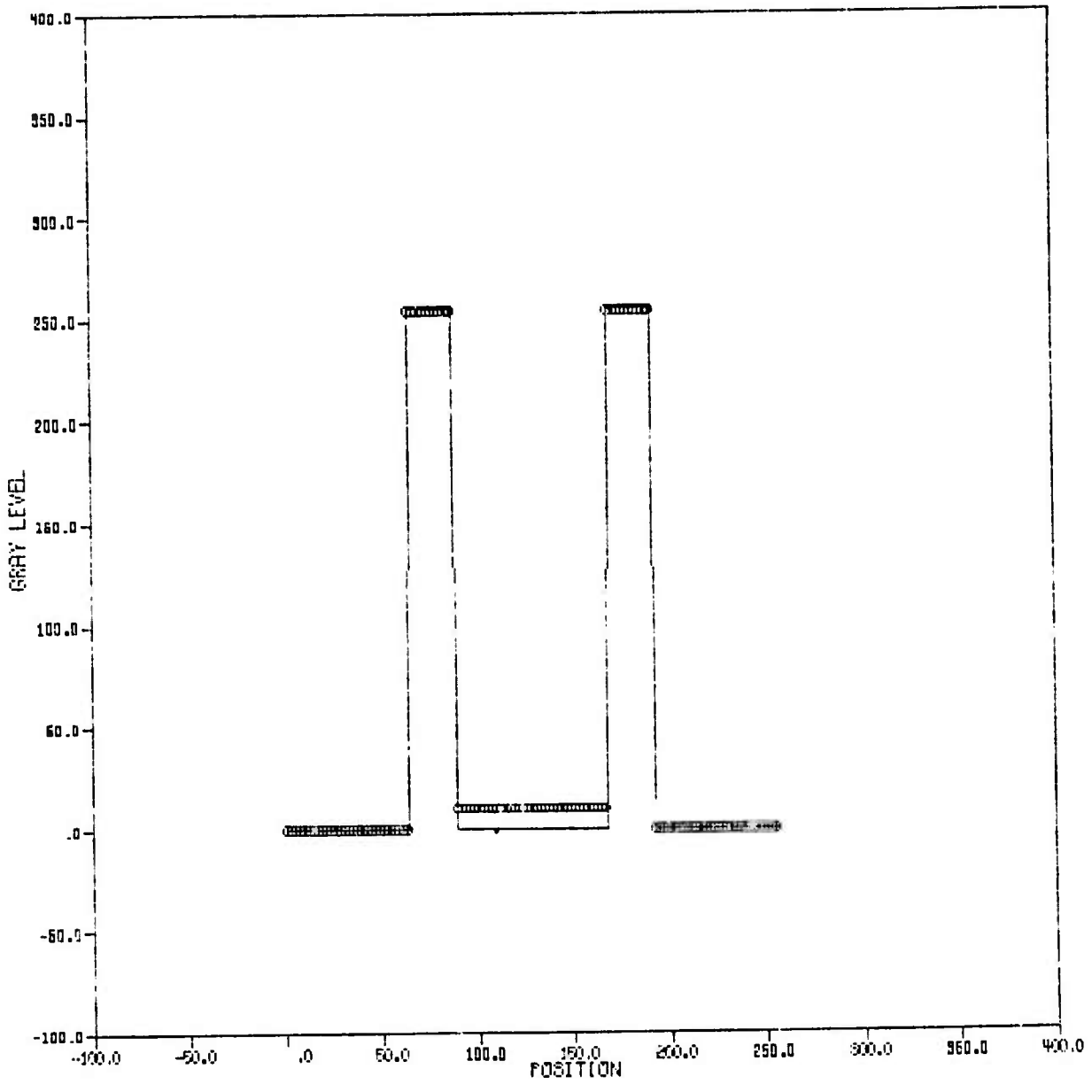


Fig. 5 Actual and Estimated Scan Line

A STUDY OF DIGITAL STRAIGHT LINES

G. Y. Tang and T. S. Huang

This report consists of four sections. In section I we investigate some fundamental properties of the digitization of a straight line. In section II we look into a special class of straight lines whose slopes are rational. In section III we develop a scheme to approximate any straight line by a rational-slope straight line such that within the scope of the digital picture they give the same digitization. In section IV we develop techniques to recognize and identify the digitization of a straight line.

Section I

The digitization scheme used in this paper is the standard grid-intersection scheme for chain coding which was first developed by Freeman [1]. Given a coordinate grid superimposed on the straight line, the grip point nearest to the crossing becomes a point of the digitization of that line. (In the case the crossing is exactly midway between two grid points, we shall assume for concreteness that we always round down to the one having smaller value.) The digitization points can then be connected by straight line segments called elements. There are only eight possible directions, numbered from 0 to 7 in counterclockwise sense. 0 is corresponding to the element lying in +X direction. Those elements have only 2 lengths, 1 and 2. A sequence of the elements is called a chain.

Some observations about the properties of the digitization of a straight line are outlined as follows [2].

1) When the straight line to be digitized has its slope strictly between -45° and 45°, its digitization depends only on its crossings of the vertical grid lines. A simple and efficient method to generate the chain of a straight line follows immediately from this observation. A line is defined by

$$Y = mx + Y_b \ , \ 0 < m \leq 1 \ .$$

The i^{th} Freeman Code is found by $Y_i - Y_{i+1}$, where $Y_i = [mi + Y_b + 1/2]$, [X] indicating the greatest integer smaller than X.

2) Likewise for a straight line with slope between 45° and 135°, only horizontal crossings play a role in the digitization of that line.

3) For the digitization of a straight line, there are at most two types of elements present, and these can differ only by unity, modulo eight. One example is the straight line with slope between 0° and 45°, only 0 and 1 can be presented. Particularly for slope equal to 7/36 its Freeman Code is

0001000100010001000100010000100 01... .

4) One of the two elements mentioned in (3) always occurs singly. In the above example it is obvious that 1 occurs singly. If we look into this property further, we will see that for straight lines lying between 0° and 26.5°, 1 occurs singly and for those lines lying between 26.5° and 45°, 0 occurs singly.

5) Successive occurances of the element occuring singly, as mentioned in (3) are as uniformly spaced as possible. Considering the same example again, we can see the spaces between two adjacent 1's are

33333343333334333334...

where 3 indicates three zeroes between them and 4 indicates four zeroes between them. We may as well call each successive occurance of 0's a run. Then there are at most two run lengths and they are consecutive integers. One of these two runs will always occur only once at a time. The other one of course will occur in runs. And furthermore, for the run length that occurs in runs,

these runs can themselves have only two lengths, which are consecutive integers and so on. An example is for a line of slope 14/39. The chain code is

0010010010010100100100100101001001001 01...

In terms of run length, we will have

222212222122221...

So, we can see the run of length 1 occurs singly and the run length of the other run are two consecutive integers, 4 and 3. 3 occurs singly.

6) Is there a necessary and sufficient condition that a digital arc is the digitization of a straight line? If we can find a yes answer to it, can we derive an efficient necessary and sufficient testing algorithm for a digital arc? Rosenfeld answers the first question in terms of the chord property?

Let a, b be points of a digital arc D, and let $\overline{ab}$ denote the real line segment between a and b. We say that $\overline{ab}$ lies near D if for any real point (x,y) on $\overline{ab}$, we can find a grid point (i,j) on D such that $\max[|i-x|, |j-y|] < 1$. We say that D has the chord property if and only if for every pair of a, b in D, the chord $\overline{ab}$ lies near S. The following two statements can thus be proved. (I) The digitization of a straight line segment is a digital arc and has the chord property; (II) If a digital arc has the chord property, we can find a straight line segment such that the digital arc is the digitization of that straight line. It is noticed in the sequal that the digitization of any curve lying on a certain straight strip will generate the same digital arc, according to our grid-intersection scheme.

Section II

In this section we put our attention on a subset of a straight line whose slopes are rational numbers. A rational number is a real number which can be

characterized by two integers p,q (no common factor between p and q). We are going to see a sort of periodic properties in this subset of straight lines. We also are going to examine the effect of Y intersection.

A Freeman chain code F is said to be periodic if we can find a subchain T of F such that $F = T^{I}R$, $I = 1,2,\ldots,R$ is a subchain of T. In order to have this definition actually significant, we assume $|F| >> |T|$ where $|F|$ is the length of F and $|T|$ is the length of T. In the following discussion, we assume the length of Freeman Code is arbitrarily long. We also say that two Freeman chains are cyclically equivalent if we can obtain one from the other by a certain number of cyclic shifts. Each cyclic shift is achieved by removing the leftmost code from the chain and to place it next to the rightmost code on the chain. The number of shifts needed between two cyclically equivalent chains is called the displacement between the two chains. Obviously the displacement is always smaller than $|T|$.

A real straight line with rational slope is characterized by the equation:

$$Y = p/q\ X + Y_b$$

p,q are integers with no common factors and q is not zero. The Freeman chain associated with the digitization has the following properties.

1) The Freeman chain is periodic with period T. So in order to investigate the relation between the slope and the Freeman chain, we need only to observe one period. Notice that we need only to consider lines between 0^{o} an 45^{o} without loss of generality. Referring to (5) in section I, "spaced as uniformly as possible", Brons [3] proposed a structural algorithm to construct a period from p and q. The algorithm is

(1) Read p and q under the assumption $0 < p < q$; p,q are integers.

(2) Reduce p|q if possible, call the new value p_o and q_o.

(3) Calculate $r_o = q_o - p_o$. (There are r_o symbols 0 and p_o symbols 1 in a period.)

(4) Let $c_l = \text{Max}(P_l, r_l)$; call symbols occuring c_l times C_l.

(5) Let $t_l = \text{Min}(P_l, r_l)$; call symbols occuring t_l times D_l.

(6) If $t_l = 1$, generate the period $C_l^{c_l} D_l$ and stop the program.

(7) Calculate $n_l = [c_l/t_l]$.

(8) Calculate $P_{l+1} = c_l - t_l n_l$.

(9) Calculate $r_{l+1} = t_l - P_{l+1}$.

(10) Generate r_{l+1} subperiods $C_l^{n_l} D_l$, call them B_{l+1}.

(11) Generate P_{l+1} subperiods $C_l^{n_{l+1}} D_l$, call them A_{l+1}.

(12) Go to step 4.

2) There is no displacement between the following two lines

$$L_1: \quad y = (p/q)X + Y_b$$

$$L_2: \quad y = (p/q)X + (Y_b + N) \ , \ N = \ldots -2,-1,0,1,\ldots$$

3) We now look into the displacement between two lines with the same slope but differing in y intersection by the amount of 1/q. These two lines are characterized by

$$L_1 : \quad y = p/q \ X + Y_b$$

$$L_2 : \quad y = p/q \ X + (Y_b + 1/q).$$

The amount of shift of the corresponding Freeman chains must meet the requirement

$$\Delta y = p/q \ r_1 + 1/q = n$$

where r_1 is an integer, n is an integer

$$r_1 = \frac{nq-1}{p} , \ldots\ldots (1)$$

n is the smallest one. For example, the Freeman code for p/q = 7/36, y_b = 0.3333 is

1000001000010000100001000010000100001 ..... .

The difference is 5 which can also be obtained by plugging in (1) with n = 1, q = 36, p = 7.

4) It is noticed that without loss of generality, only straight lines with y intersections in an interval of length one are needed to be considered. For convenience, we pick up the interval $[\frac{1}{2},\frac{3}{2}]$. Recall that the Freeman chain of a straight line with rational slope p/q has period q. So there are at most q different chains corresponding to that slope. It is found that each of the following intervals determines a chain

$$(\tfrac{1}{2} , \tfrac{1}{2}+\tfrac{1}{q}) , (\tfrac{1}{2}+\tfrac{1}{q} , \tfrac{1}{2}+\tfrac{2}{q}) , \ldots , (\tfrac{1}{2}+\tfrac{1}{q} , \tfrac{1}{2}+\tfrac{i+1}{q}) , \ldots , (\tfrac{3}{2}-\tfrac{1}{q} , \tfrac{3}{2}) .$$

The Freeman code of a straight line with y intersection in the interval $(\frac{1}{2} , \frac{1}{2}+\frac{1}{q})$ is the one got by the structural algorithm. The one in $(\frac{1}{2}+\frac{1}{q} , \frac{1}{2}+\frac{2}{q})$ is obtained by shifting r_1.

All the lines with intersection in the strip defined by two parallel lines y = p/q X + a and y = p/q X + a + 1/q should have the same digitization. More than that, all curves lying in that strip should have the same digitization too. Thus there are many curves corresponding to one digital curve.

Section III

In Section II, we have outlined a few properties of the digitization of a subset of all straight lines. That particular subset is the set of all

straight lines with rational slope. In this section, we are going to look into real-slope lines. Our goal is to find an algorithm such that for any real slope we can find a rational slope such that the digitization of these two straight lines are identical for the first certain number of left-most codes. Assume the straight line with a real slope is

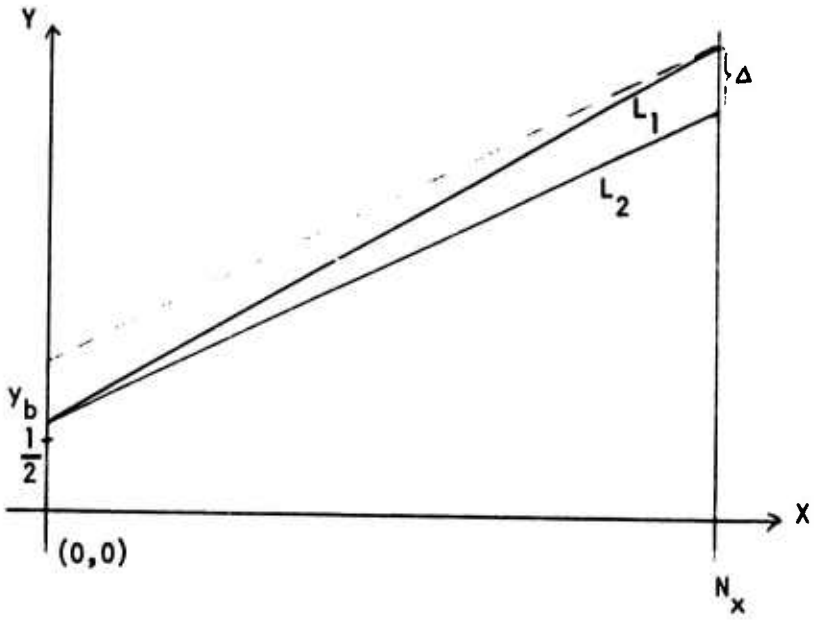
$$L_1 : y = \alpha X + Y_b , \alpha \in R, 0 < \alpha < 1, 1/2 < Y_b < \frac{3}{2} ,$$

the one with rational slope is

$$L_2 : y = \frac{P}{N_x} X + \frac{N}{N_x} + \frac{1}{2} ,$$

N is an integer such that

$$0 < Y_b - \frac{1}{2} - \frac{N}{N_x} , Y_b - \frac{1}{2} - \frac{N+1}{N_x} > 0 , \frac{P}{N_x} < \alpha :$$



From Fig. 1, we can see that

$$\Delta = (\alpha N_x + y_b) - (\frac{P}{N_x} N_x + \frac{1}{2} + \frac{N}{N_x})$$

$$= \alpha N_x - P + (y_b - \frac{N}{N_x} - \frac{1}{2})$$

Figure 1 Rational approximation of a real slope

Obviously, we have $Y_b - \frac{N}{N_x} - \frac{1}{2} < \Delta < \frac{1}{N_x}$. So, we got $p \geq \alpha N_x - \frac{1}{N_x} + (Y_b - \frac{N}{N_x} - \frac{1}{2})$ and $\alpha N_x \geq p$, p is an integer.

An algorithm to find the rational number is hereby constructed.

1. Read $\alpha \cdot Y_b$, N_s (a constant for increment).
2. Set $N_x = N_o$ (N_o is an initial value).
3. Find N such that $0 < Y_b - \frac{1}{2} - N/N_x$ and $Y_b - \frac{1}{2} - \frac{N+1}{N_x} > 0$.
4. Set $N_x = N_x - N_s$.
5. Set $PP = (\alpha N_x + Y_b) - (\frac{N+1}{N_x} + \frac{1}{2})$.
6. Set $P = [PP] + 1$.
7. Is $P \leq \alpha N_x$? If it is yes, then stop, otherwise $N_x = N_x + N_s$, go to 2.

Usually, N_s is set to be 1. This algorithm will give us:

α = 0.15315606 , P=27, Q=111, and P/Q=0.15315315.

α = 0.23648940 , P=35, Q=148,,and P/Q=0.23648609.

α = 0.81982273 , P=91, Q=111, and P/Q=0.81981982.

Section IV

From Section I to Section III we have investigated various inherent properties of a straight line. In this section, we will try to combine all the knowledge we have and to see if we can use it to recognize a digital arc, i.e., to determine whether there is a straight real line corresponding to that digital arc.

One immediate application of the properties to do recognition is that if we know already that a string of chain codes is a period, then we can proceed as following:

1. The length of that string, let it be q.
2. The number of the elements which occur singly, let it be P.
3. Using the structural algorithm to generate a period of chain code, which is called CC.

4. Comparing the input string with the new chain CC to see if they are cyclical equivalent.

5. If they are cyclical equivaient, then the string is the digitization of a straight iine.

The foregoing aigorithm is very limited. But we can extend the concept to a more general case. Suppose we have a string of Freeman code at hand. We want to decide if it is the digitization of a straight line. The new one would turn out rather lengthy in computing time. In order to alieviate the tragic time consumption, some simpie a priori test may be performed. Exampies of the a priori test are:

(1) Are there only two elements one of which occur singiy?

(2) Are there oniy two run iengths which are consecutive integers? And one of them occur singiy?

(3) Are the run lengths of the runs that occur in runs two consecutive integers?

If a digitization obtained aii "yes" in the a priori test, then we can find the ratio of the number of the element ocurring singiy to the length of the input string. This ratio, say, is p/q. We can thus know that if it is a digitization of a straight line, the slope must be in the intervai enclosed by (p+1)/q and (p-i)/q. So the rational equivaient line must have a siope between (p+1)/q and (p-1)/q. Unfortunately, there are infinite but countable number of rationai numbers lying between (p+1)/q and (p-i)/q. If we got to test aii of them, it would be impracticai. How to reduce the number of testing to be finite is stiii unsolved. But a simpie heuristic solution might be applied by setting an upper limit of the number of testing. If we go beyond the iimit, we would claim heuristicaiiy that we can never find a rationai slope to give an equivaient digitization to the reai siope line. From the argument in

section III, this is a contradiction. Hence we assert that it is not a digitization of a straight line. The Freeman code of the curve,

$$y = 7/36\ X + 0.50926 + \frac{1}{3 \times 120} X^2$$

which only slightly deviates from a straight line $y = 7/36\ X + 0.50926$, when X is not very big, is

0000100001000011000010010100001l .... .

We can see that it will be rejected in the a priori testing stage. But if we knew only the first 15 elements, i.e., 000010000100001, we would have $p/q = 3/15$. So the two limits are 4/15 and 2/14. Obviously, $2/15 < 7/36 < 4/15$. If we could test enough rational numbers between 2/15 and 4/15 we would find 7/36 and we would assert that it is a digitization of a straight line, and indeed it is.

REFERENCES

[1] H. Freeman, "On the Encoding of Arbitrary Geometric Configurations," IRE Trans. Electron Comp., Vol. EC-10, No. 2, June 1961.

[2] A. Rosenfeld, "Digital Straight Line Segment," IEEE Trans. on Comp., Vol. C-23, No. 12, December 1974.

[3] R. Brons, "Linguistic Methods for the Description of a Straight Line on a Grid," Computer Graphics and Image Processing, 3, pp. 48-62, 1974.

IMAGE SEGMENTATION BY UNSUPERVISED CLUSTERING

M. Y. Yoo and T. S. Huang

I. Introduction

We have shown results of image segmentation based on the feature pair: local mean, local standard deviation; and various clustering algorithms (simple merging by Euclidean distance, graph theoretic clustering) in the last three reports. Our experience has confirmed that it is more important to find better features which give significantly separated clusters than to develope an efficient clustering algorithm. We now show results of image segmentation based on other feature pairs.

II. Experimental Results

We will describe several feature pairs and show some results based on them.

(1) Local mean and standard deviation:

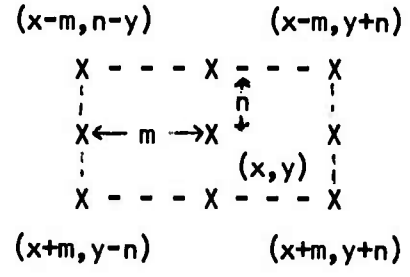


Figure 1 Local Window Used for Local Operations

Let $P(x,y)$ be the picture array then the local mean $M(x,y)$ and the local standard deviation $S(x,y)$ at (x,y) based on the local windows of size $m \times n$ are

$$M(x,y) = \frac{1}{(2m+1)(2n+1)} \sum_{\mu=x-m}^{x+m} \sum_{\sigma=y-n}^{y+n} P(\mu,\sigma)$$

$$S(x,y) = \frac{1}{(2m+1)(2n+1)} \sum_{\mu=x-m}^{x+m} \sum_{\sigma=y-n}^{y+n} \{P(\mu,\sigma) - M(x,y)\}^2 \; ^{1/2}$$

The size of the window m x n may be chosen appropriately for the images to be considered. m=n=1 has been used.

(2) Local median and modified deviation:

Let $D = \{(\alpha,\beta): \alpha = x-m, \ldots, x+m; \beta = y-n, \ldots, y+n\}$

then the local median and the modified deviation are

$$MED\ (x,y) = \underset{(\alpha,\beta)\ \varepsilon\ D}{\text{median}}\ P(\alpha,\beta)$$

$$DEV\ (x,y) = \underset{(\alpha,\beta)\ \varepsilon\ D}{\text{median}}\ |\ P(\alpha,\beta) - MED\ (x,y)|$$

The feature plane based on this measure is given in Fig. 3 (computer print out) and Fig. 4 (Gould print out). The Fig. 5 and Fig. 6 show the filtered feature and the corresponding image, respectively.

(3) Local minimum and local maximum:

$$MIN\ (x,y) = \underset{(\alpha,\beta)\ \varepsilon\ D}{\text{minimum}}\ P(\alpha,\beta)$$

$$MAX\ (x,y) = \underset{(\alpha,\beta)\ \varepsilon\ D}{\text{maximum}}\ P(\alpha,\beta)$$

Figures 7, 8, and 9 show feature plane and filtered images based on local minimum and local maximum.

(4) The number of "jumps" and the average magnitude of the jump:

We compare two adjacent points (in all directions) in the local window and if the gray level change is greater than the preassigned threshold value we assumed there is a jump.

We take the total number of jumps in the local area as a feature 1 and the average magnitude of a jump as a feature 2.

feature 1 = total number of jumps in the local image of size m x n

feature 2 = the average magnitude of a jump in the area

Figures 10 and 11 show the distribution of features for two different threshold values and the filtered features and the corresponding images are shown in Figures 12 and 13.

This model can be modified a little bit in counting the total number of local jump.

In feature 1 we counted one jump whenever the change in gray levels of adjacent points exceeds the threshold value, but in the modified version two successive jumps in the same direction (up or down) are counted as one jump.

Figure 14 is the feature based on this modified version and Figure 15 and 16 show the filtered feature and image correspondingly. In all cases we used m=n=1.

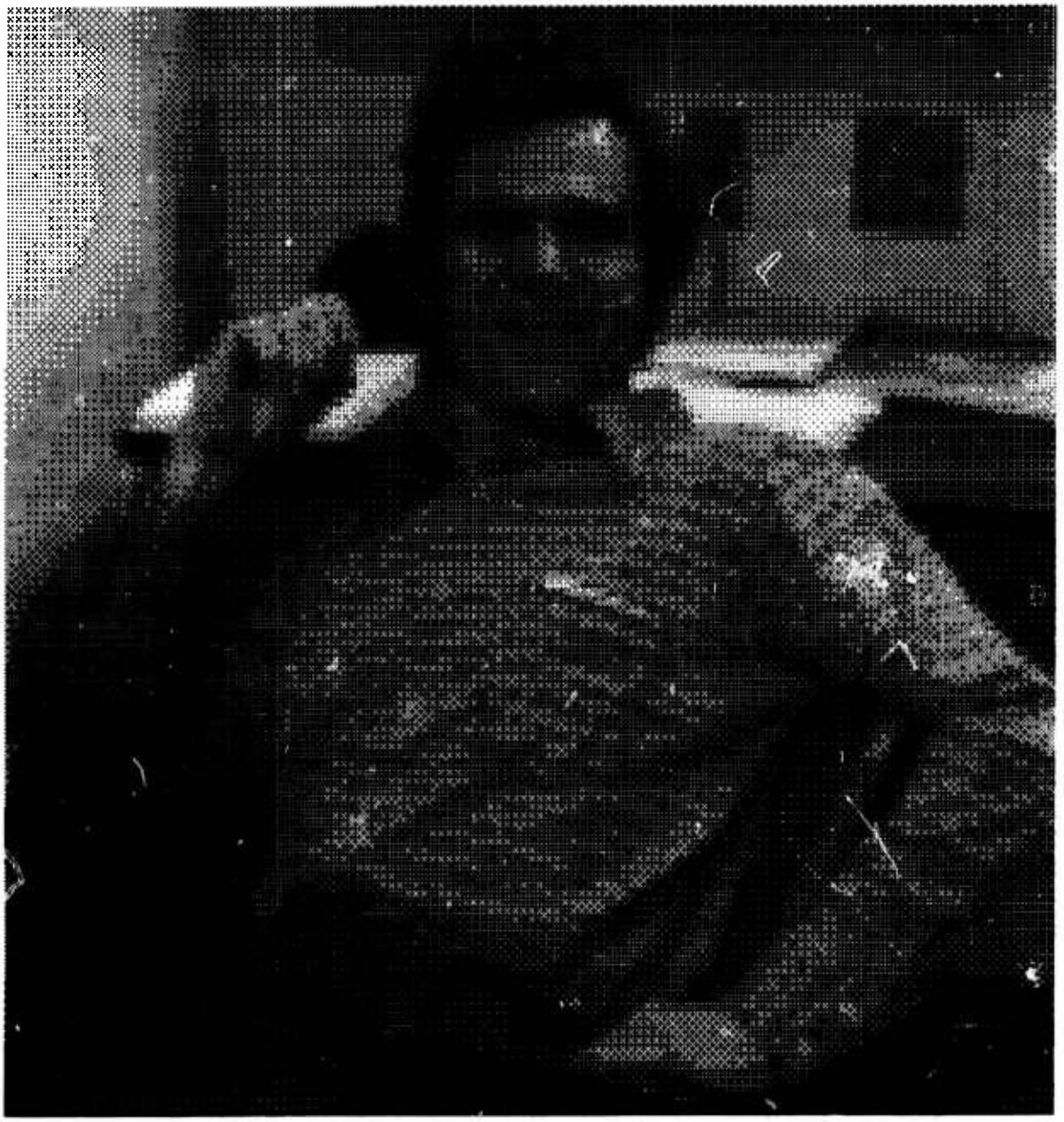


Figure 2 Original Image

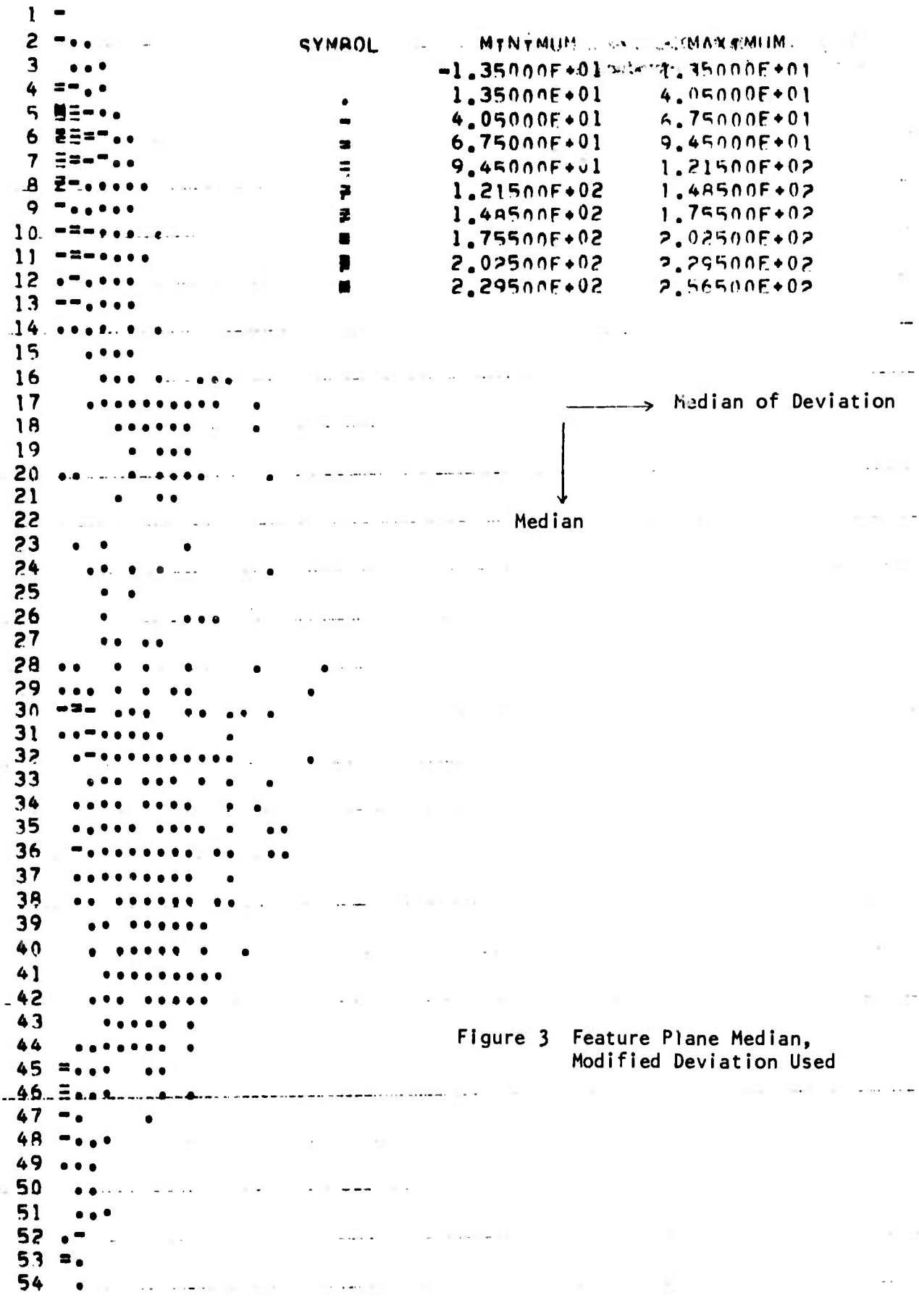


Figure 3 Feature Plane Median, Modified Deviation Used

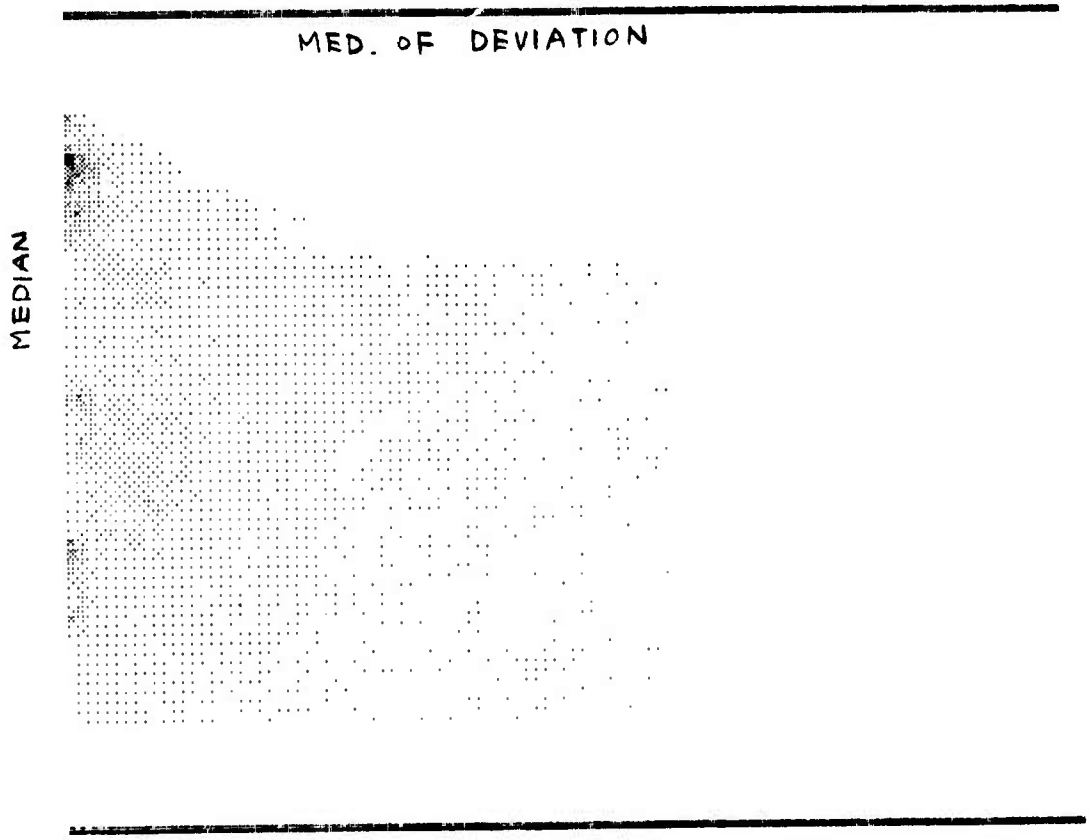


Figure 4 Feature Plane Same as Fig. 2 (Gould Print)

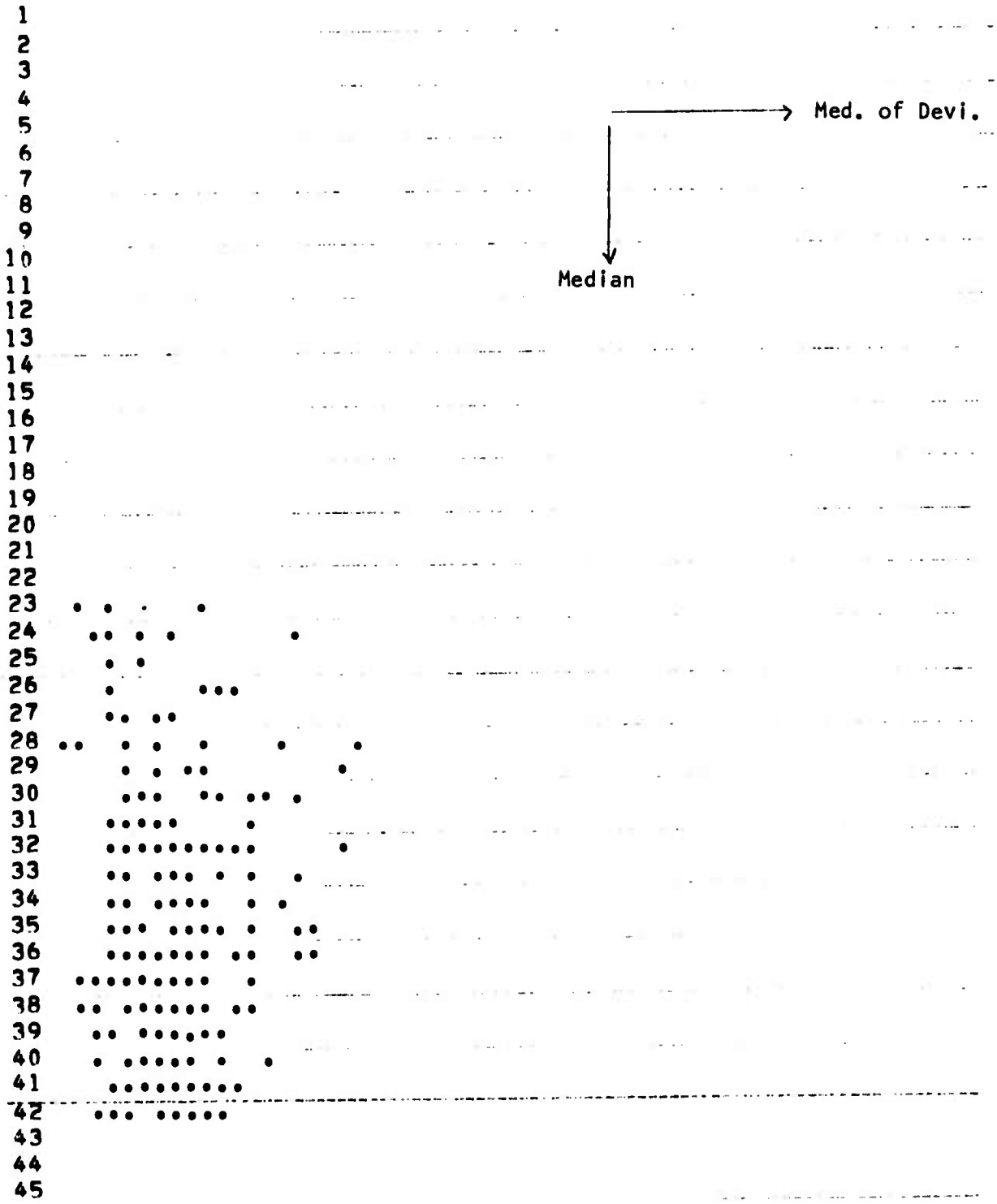


Figure 5 Feature Plane (Filtered) Median (X), Modified Deviation (Y) Used

Figure 6 Filtered Picture (Note Fig. 4)

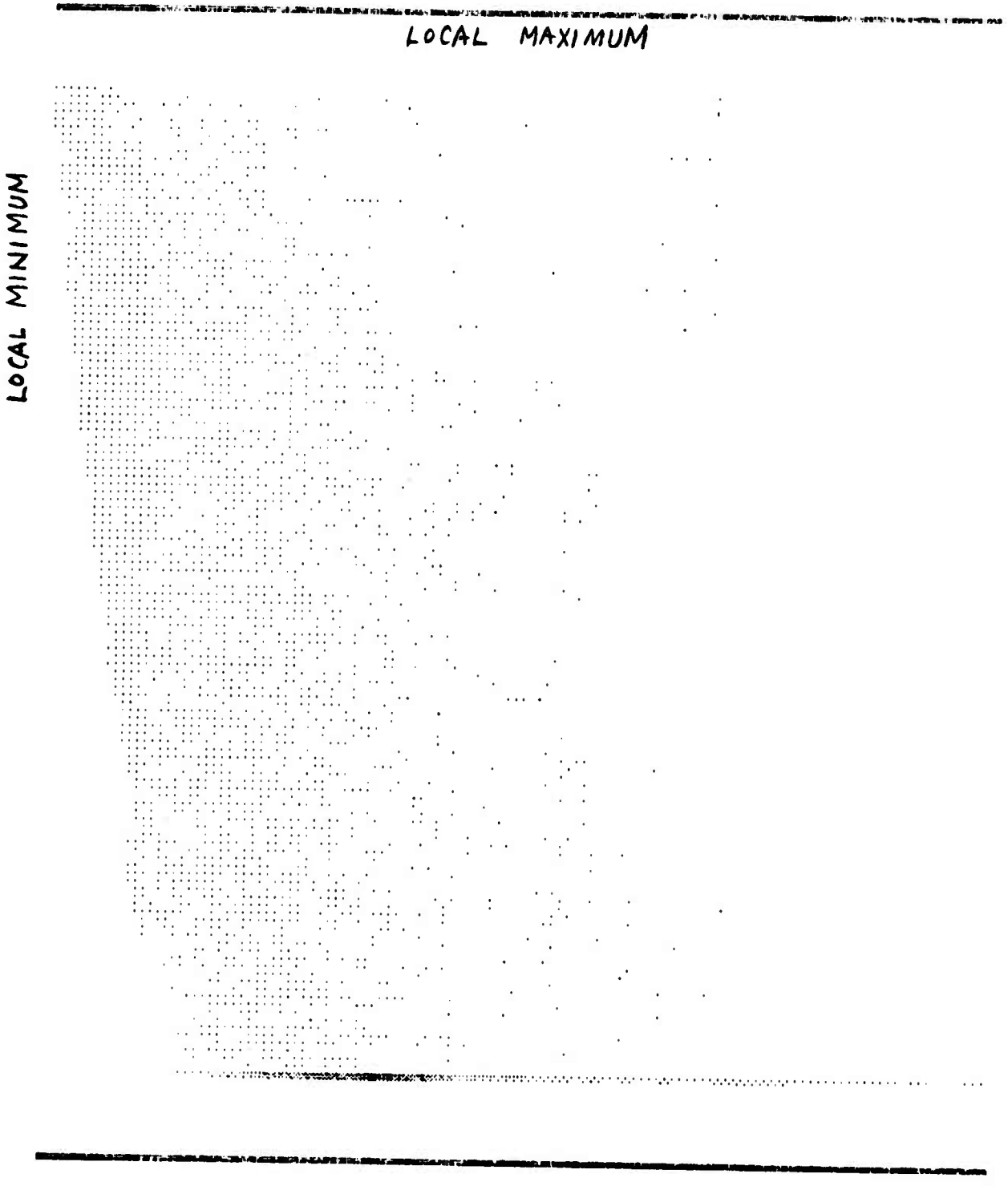


Figure 7 Feature Plane, Local Minimum, Local Maximum Used

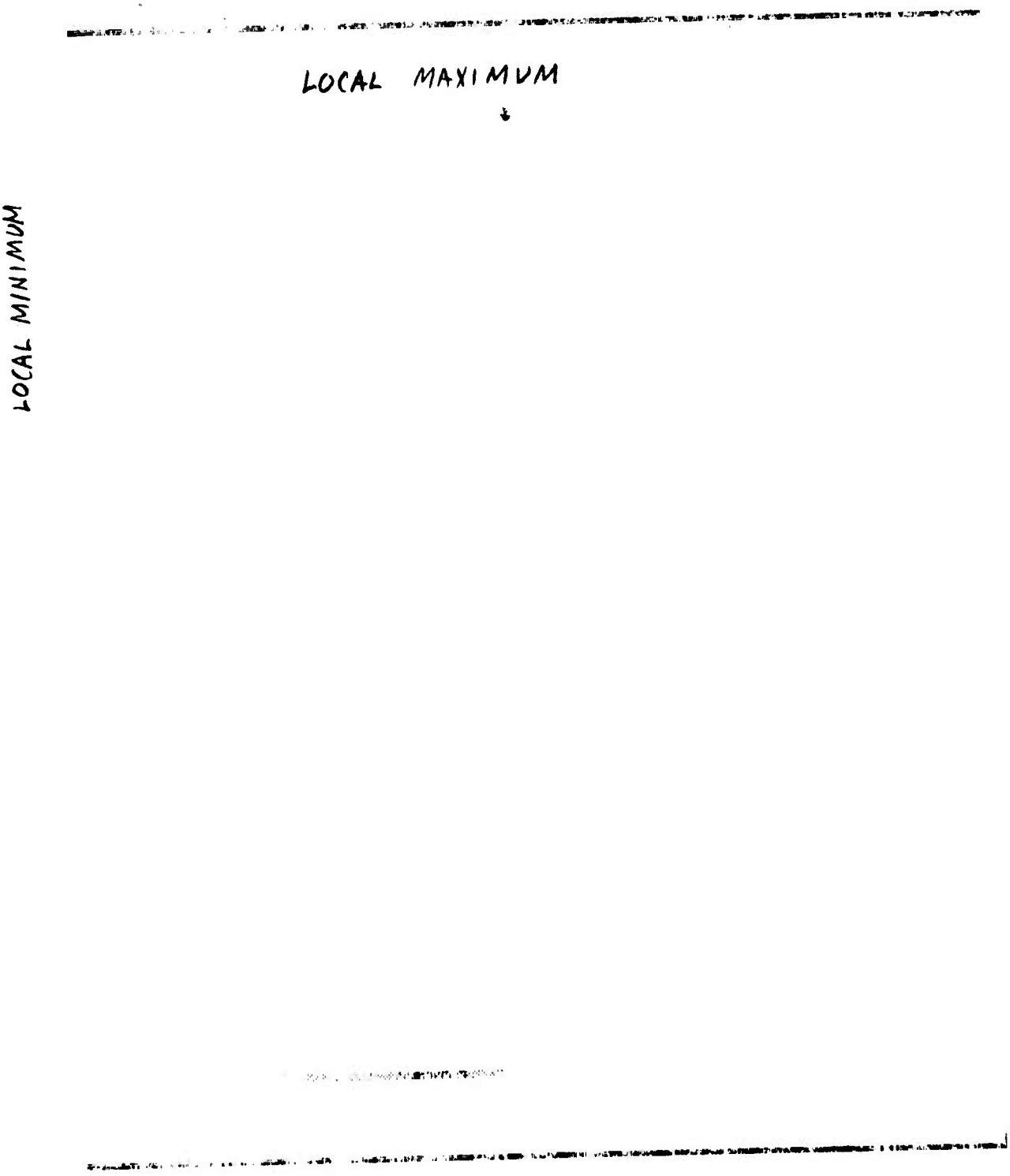


Figure 8 Filtered Feature (Note Fig. 7) Local Minimum, Local Maximum Used

Figure 9 Filtered Picture (Note Fig. 8)

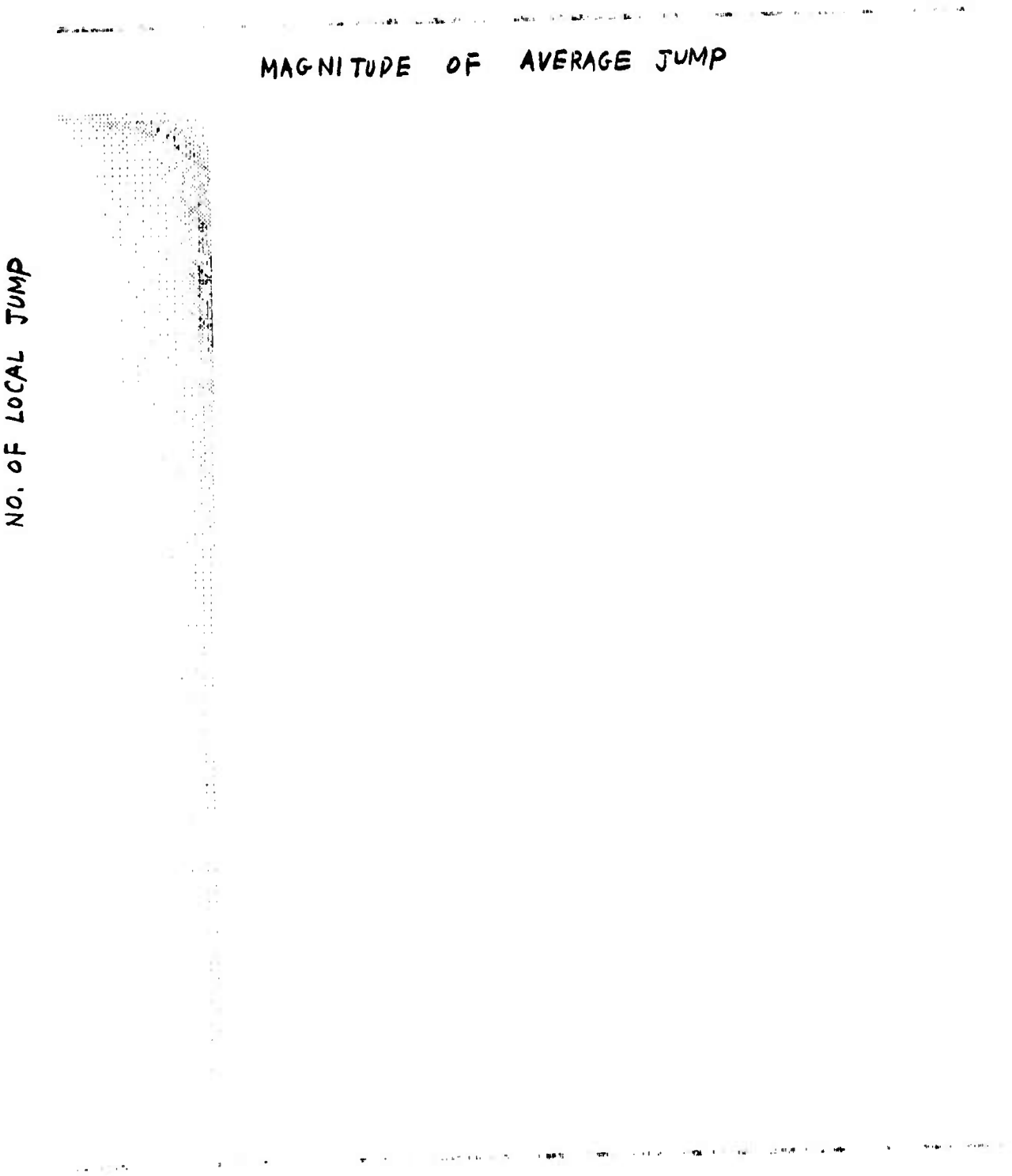


Figure 10 Feature Based on Local Jump
Threshold = 20 Used

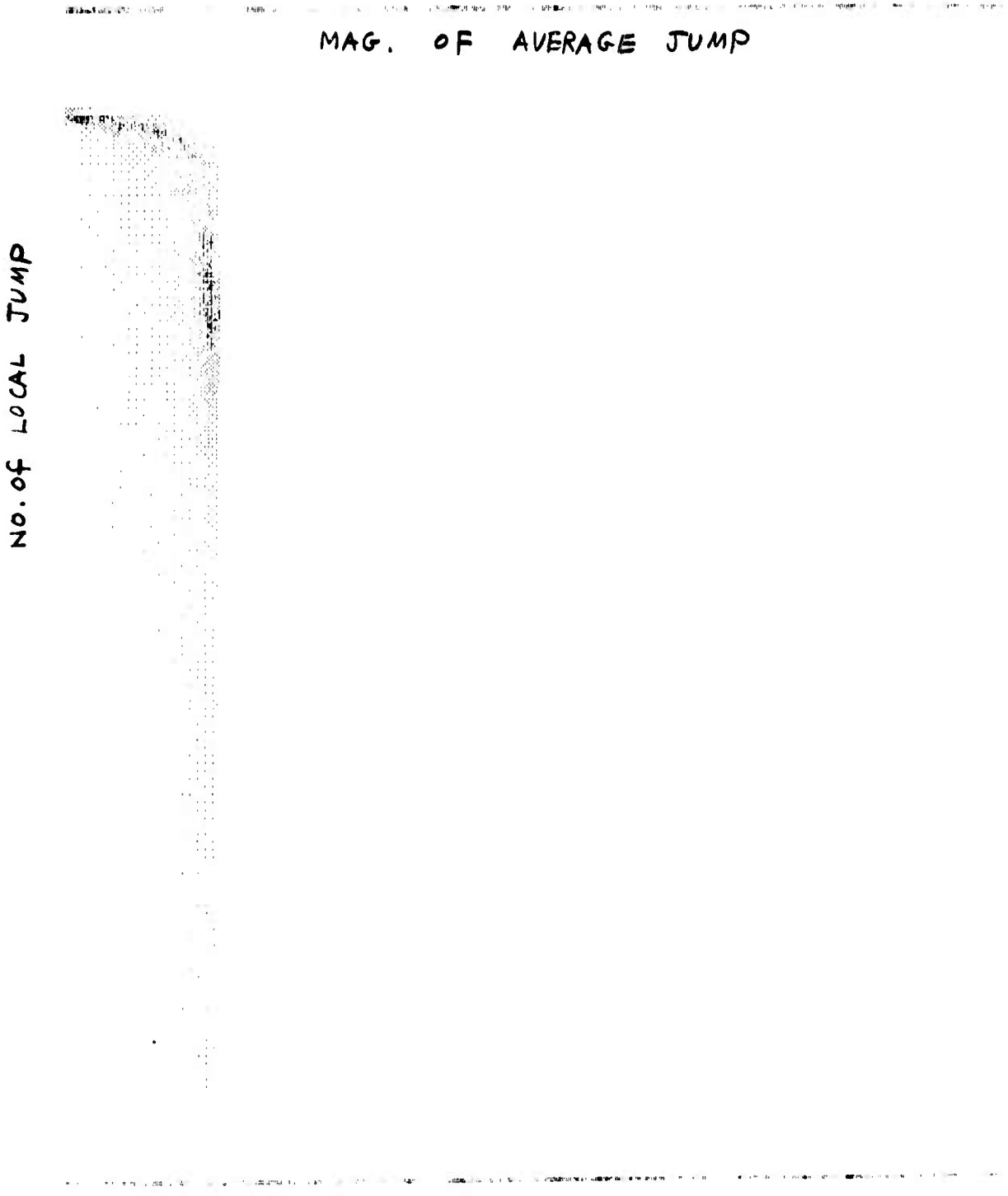


Figure 11 Feature Based on Local Jump
Threshold = 40 Used

Figure 12 Filtered Feature (Note Fig. 10)

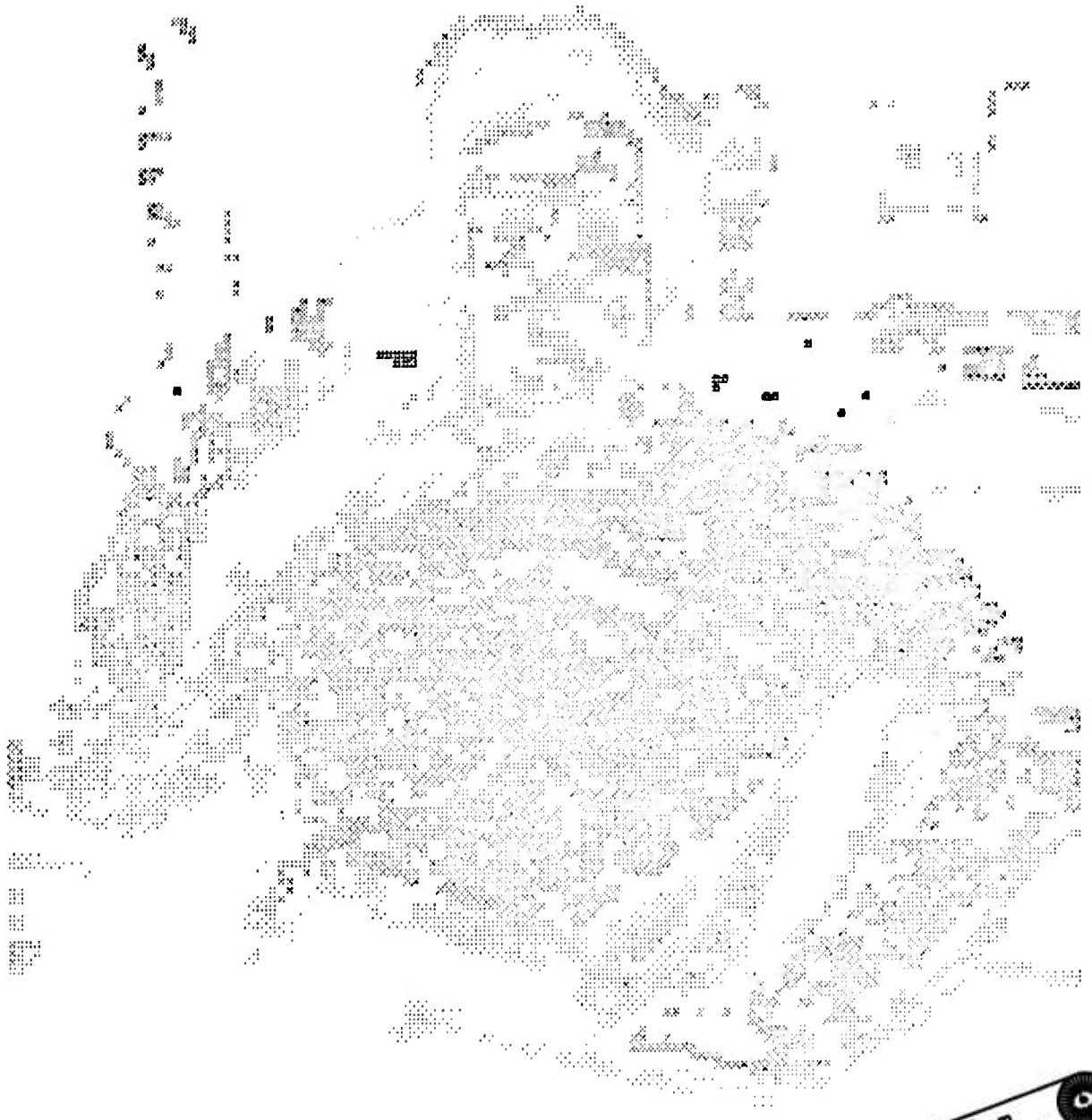


Figure 13 Filtered Picture (Note Fig. 12)

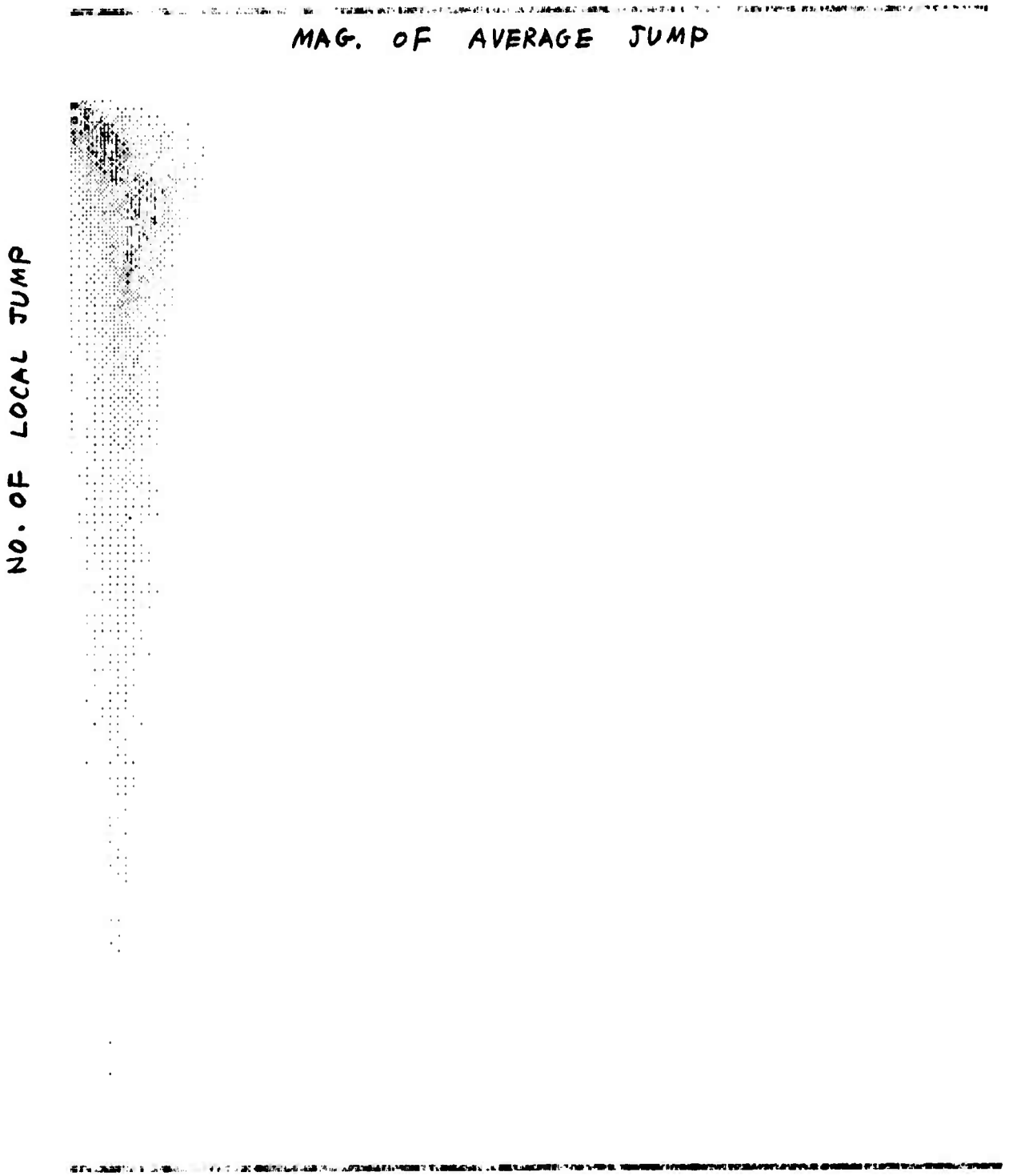


Figure 14 Modified Version of Local Jump Model Threshold = 60 Used

Figure 15 Filtered Feature (Note Fig. 14)

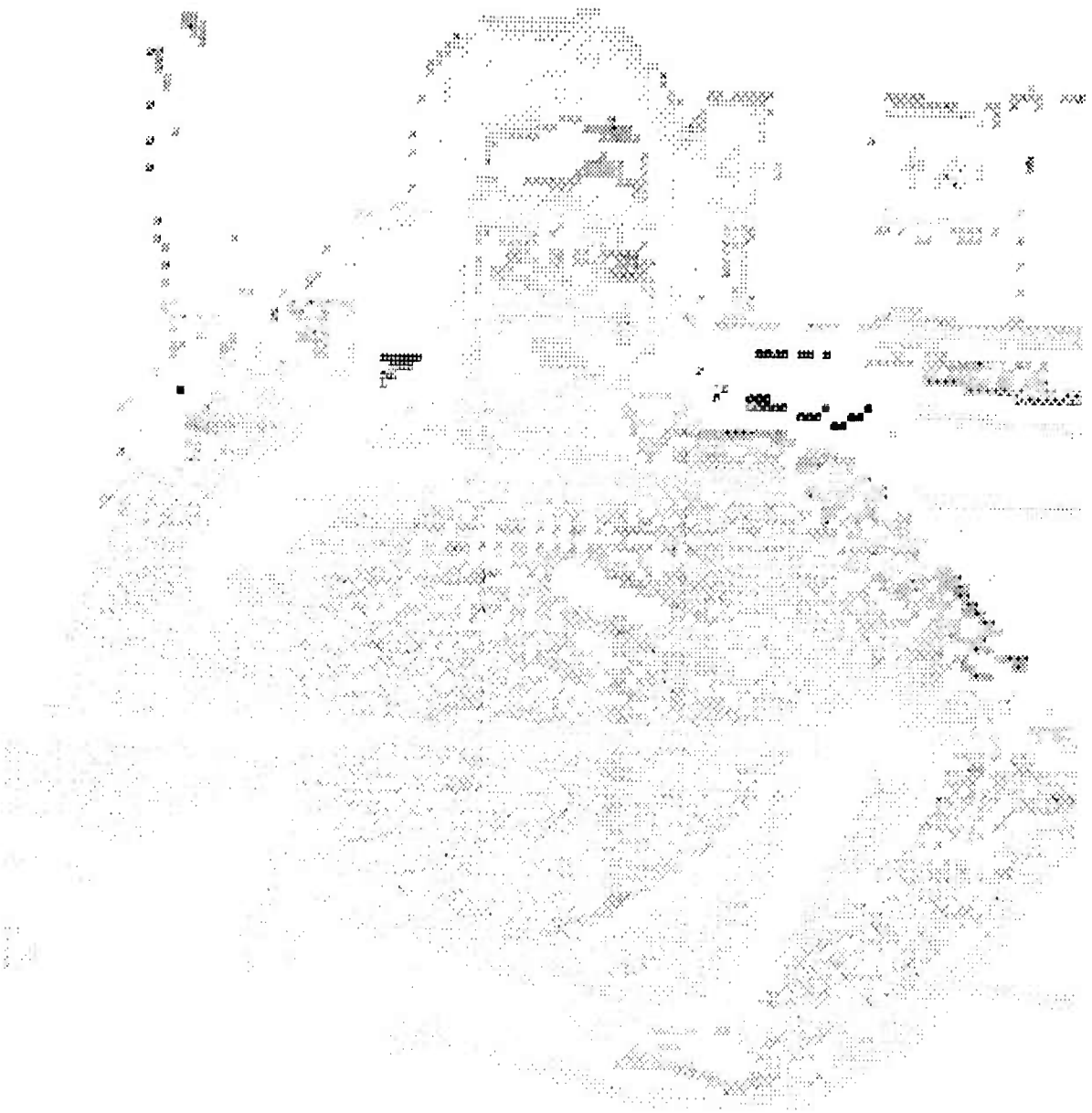


Figure 16 Filtered Picture (Note Fig. 15)

TEXTURE EDGE DETECTION AND CLASSIFICATION USING MAX-MIN DESCRIPTORS

O. R. Mitchell

I. Introduction

The detection of the boundary between two regions of differing textures is an intriguing problem. The ability to do this consistently would be a valuable tool for the broader task of image segmentation which is generally an early step in the automatic extraction of information from images.

Image texture is an ill-defined term which implies local patterned variations over a region. As an example of a texture boundary, Fig. 1 shows two texture regions, each taken from a book by Brodatz [1]. The texture on the left is cork and that on the right is wood. Also two strips in the picture have been multiplied by 0.5 to create artificial non-texture boundaries such as might be caused by shadows.

A standard method for boundary detection is to perform local averages over two adjacent windows and compare them, expecting a large difference if a boundary lies between the two windows. However, for this method to be applicable to texture boundaries some transformation must be first performed to change local texture properties to grey levels as suggested by Rosenfeld et. al. [2], [3]. This is essentially the method of texture edge detection to be described in this paper. The transformation uses a new process called "Max-Min Descriptors" [4].

II. The Max-Min Transformation

The new measure for texture analysis is based on the intuition that the important texture information for the human visual system is contained in the relative frequency of local extremes in intensity. The principal measurement in this process is the determination of the local grey level maxima and minima and their sizes along a one-dimensional scan direction. The grey level

values are first sent through a smoothing process which eliminates reversals of small amplitude, thereby retaining only the principal extrema. The smoothing algorithm is the digital equivalent of the familiar analog mechanical process known as gear backlash and it was originally described as a preprocessing method for character recognition [5].

The smoothing algorithm is described as follows: let x_k be the grey level of the $k^{\underline{th}}$ point along the scan line and let y_k be the "smoothed" value. Let T be the value of a preassigned threshold parameter. Let us start with $y_1 = x_1$ and proceed according to the algorithm shown below:

IF	THEN
$y_k < x_{k+1} - \frac{T}{2}$	$y_{k+1} = x_{k+1} - \frac{T}{2}$
$x_{k+1} - \frac{T}{2} < y_k < x_{k+1} + \frac{T}{2}$	$y_{k+1} = y_k$
$x_{k+1} + \frac{T}{2} < y_k$	$y_{k+1} = x_{k+1} + \frac{T}{2}$

Fig. 2 illustrates the way in which the smoothing process eliminates reversals of small amplitude, thereby retaining only the principle extrema. By repeating this process for several threshold settings a group of extrema can be obtained to characterize the texture. For the purpose of edge detection each extrema is labeled according to the maximum threshold at which it still exists, as shown in Fig. 3

Actually grey level densities (log intensities) are used in the extrema detection process. This changes multiplicative effects (such as lighting level or film processing) into additive changes so that very few changes occur in extrema locations and sizes. The logarithmic transformation is also similar to the human visual system's brightness sensitivity.

III. Edge Detection Results

The extrema used for edge detection were generated by first detecting local extrema by scanning in the horizontal direction and then rescanning in the vertical direction. The extrema detected in this manner from the image in Fig. 1 are shown in Fig. 4. Brightness indicates extrema size in this picture. This extrema image is the input to the edge detection algorithm. At each point two windows are placed as shown in Fig. 5. Then an edge value at that point is calculated:

$$EV = \sum_i \frac{\left| \sum_A e_i - \sum_B e_i \right|}{\sum_A e_i + \sum_B e_i + K}$$

where $\sum_A e_i$ is the total number of extrema of size i in window A and K is a constant introduced to reduce the effect of isolated extrema.

Figure 6 shows the edge values using the extrema in Fig. 4, a window size of 64 x 48, and 10 extrema levels. The edge value picture is then input to the extrema detection algorithm again with only local maxima above a fixed threshold being kept. This result is shown in Fig. 7 and is the resulting detected edge. This edge is shown superimposed on the original in Plate 1.

A more complex texture mosaic is shown in Fig. 8. The upper left section is cork, upper center is wood, lower left is pebbles, and right is magnified paper. The extrema are shown in Fig. 9, edge values using a 64 x 48 window in Fig. 10, and detected edges in Fig. 11. In Plate 2 the detected edges are shown superimposed on the original textures.

Another texture mosaic is shown in Fig. 12. It consists of 64 texture regions each 64 x 64. Plate 3 shows the detected boundaries superimposed on the original. The window size used was 32 x 32.

IV. Other Applications

Shown in Fig. 14 is a 256 x 256 x-ray image. The detected maximum extrema are shown in Fig. 15. Due to the initial logarithmic transformation, these extrema should be little affected by attenuation factors in the x-ray. Level slicing can be applied to the extrema picture to show selected aspects. For example, Fig. 16 shows only the lowest two levels of extrema. This might be useful in detecting a lung disease which shows as a texture change. Figure 17 shows edge values using a 16 x 16 window and two extrema levels. Plate 4 shows the detected edges superimposed on the original x-ray.

Shown in Fig. 19 is a 256 x 256 aerial photograph. The detected maxima extrema are shown in Fig. 20 and the detected edges using a 16 x 16 window and two extrema levels are shown in Plate 5 superimposed on the original. A real potential exists in segmenting complex pictures such as this by using a set of different window sizes and a hierarchical structure of segmented edges.

REFERENCES

1. P. Brodatz, Textures, New York: Dover Pub. Inc., 1966.

2. L. S. Davis, A. Rosenfeld, and J. S. Weszka, "Region Extraction by Averaging and Thresholding," IEEE Trans. on Sys., Man, and Cyber., pp. 383-388, May 1975.

3. S. W. Zucker, A. Rosenfeld, and L. S. Davis, "Picture Segmentation by Texture Discrimination," IEEE Trans. on Computers, pp. 1228-1233, Dec. 1975.

4. O. R. Mitchell, C. Myers, and A. Boyne, "A Max-Min Measure for Image Texture Analysis," IEEE Trans. on Computers, (in press).

5. S. J. Mason and J. K. Clemens, "Character Recognition in an Experimental Reading Machine for the Blind," in Recognizing Patterns, P. A. Kolers and M. Eden, Eds., Cambridge, Massachusetts: M.I.T. Press, pp. 156-167, 1968.

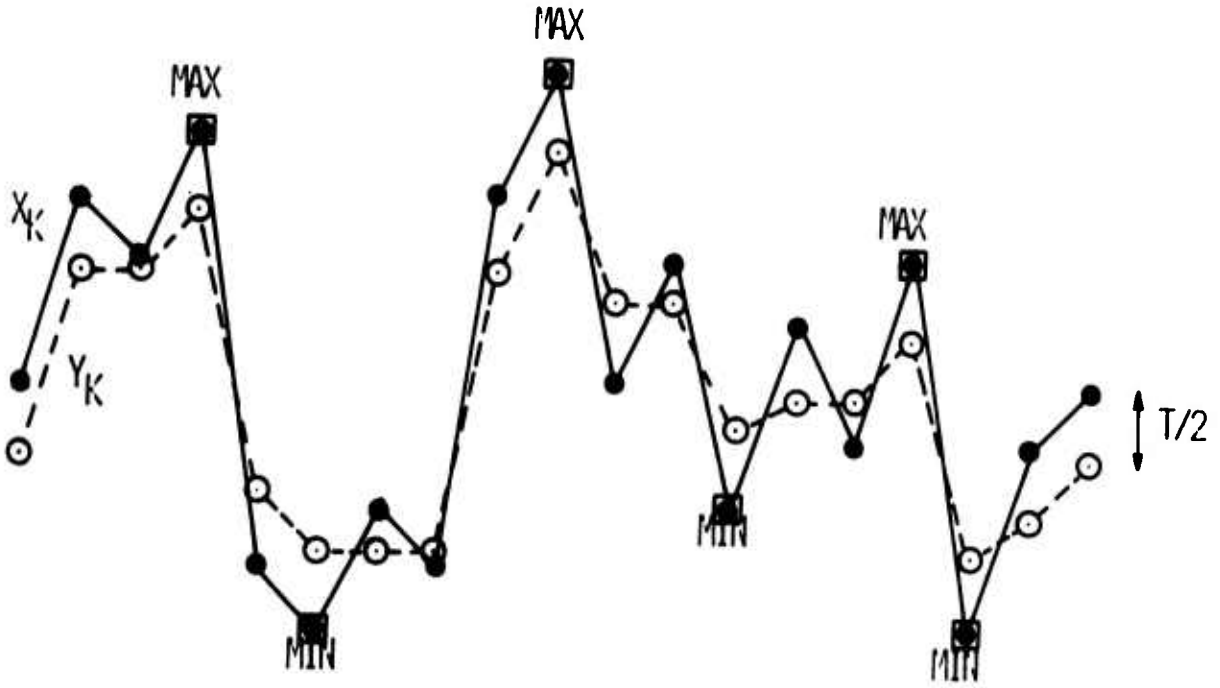


Figure 2

Dotted curve shows smoothed version of solid curve using algorithm described in the text. Local extrema remaining in smoothed curve are marked on original.

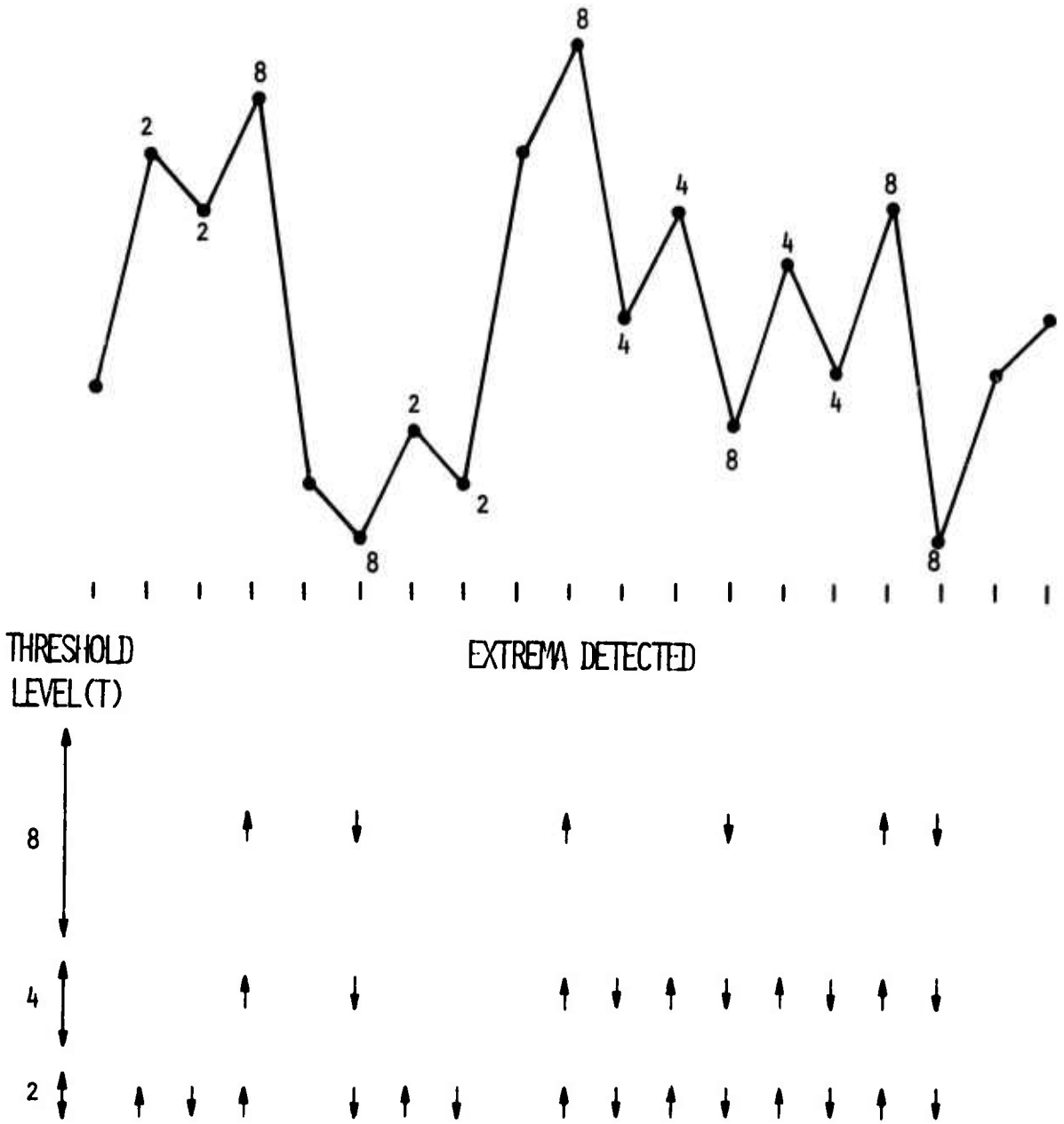


Figure 3

Each extrema is labeled according to the maximum threshold at which it still exists.

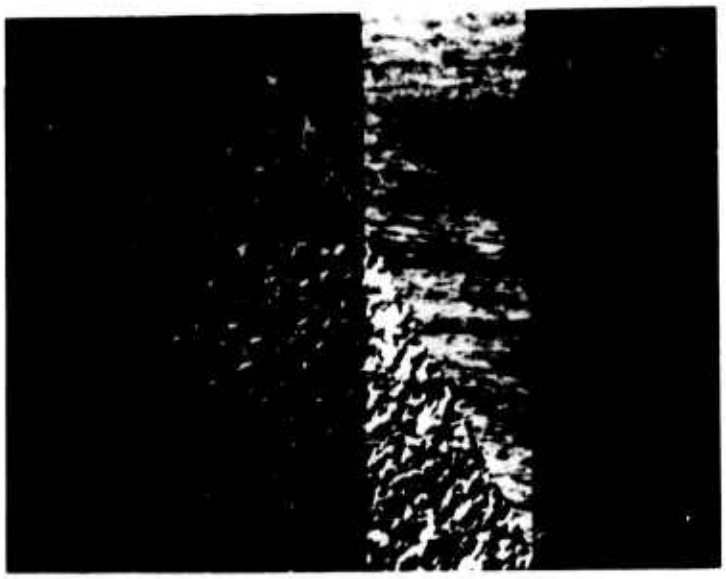


Figure 1

Two texture regions (cork and wood) with artificial boundaries created by attenuating certain areas of the picture.

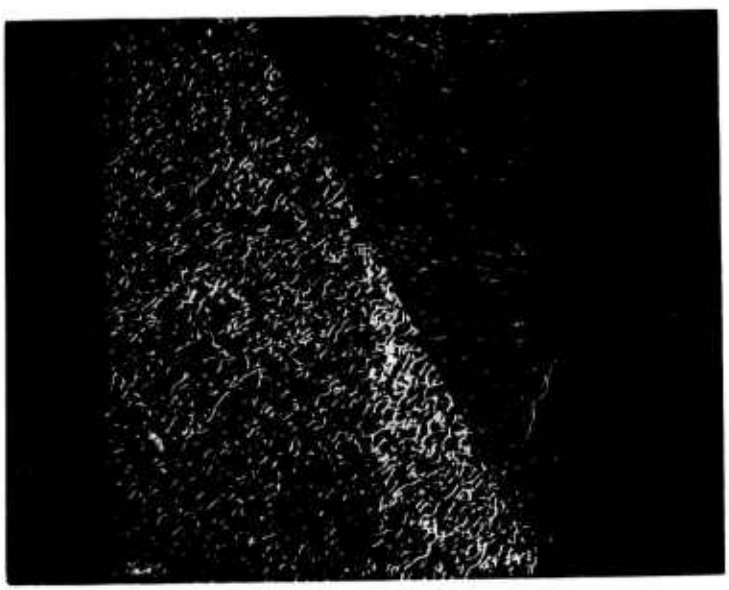


Figure 4

Extrema detected in Fig. 1. Brightness indicates extrema size. Changes in illumination cause some extrema changes due to saturation of logarithmic curve applied.

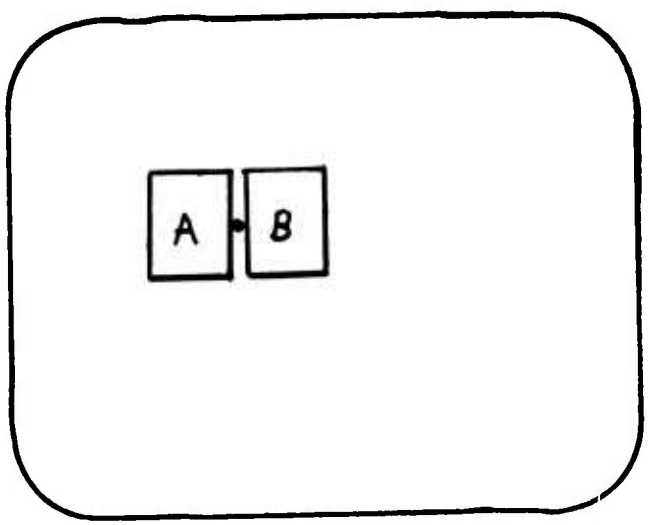


Figure 5

Window placement for each point in the picture for edge value determination.

Figure 6

Edge values from Fig. 3 using algorithm described in the text and a window size of 64 x 48.

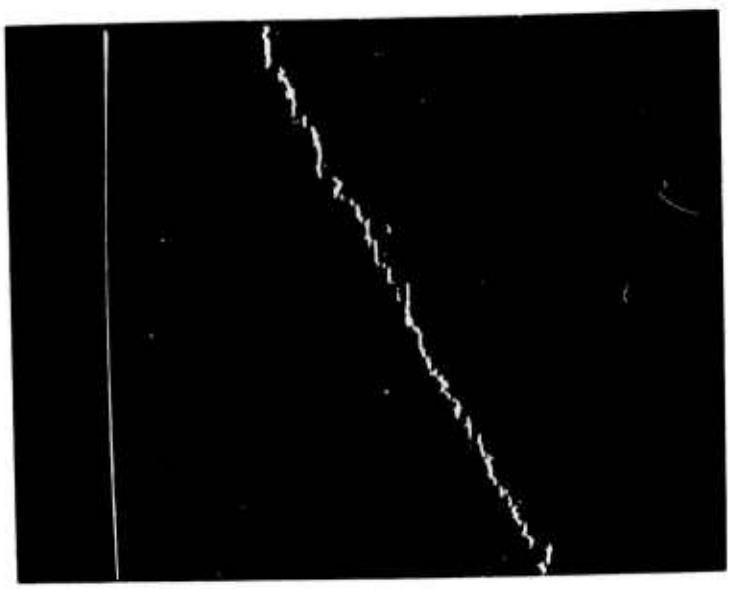


Figure 7

Texture edge detected in Fig. 1

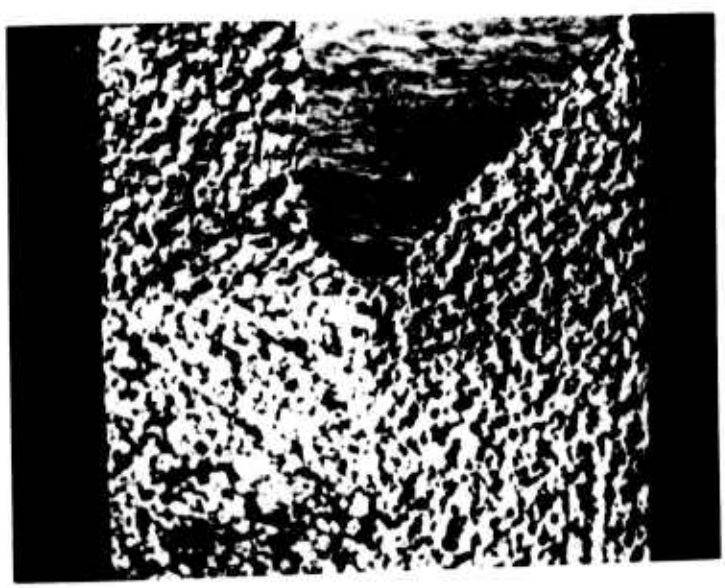


Figure 8

Another texture mosaic. The upper left section is cork, upper center is wood, lower left is pebbles, and right is magnified paper.

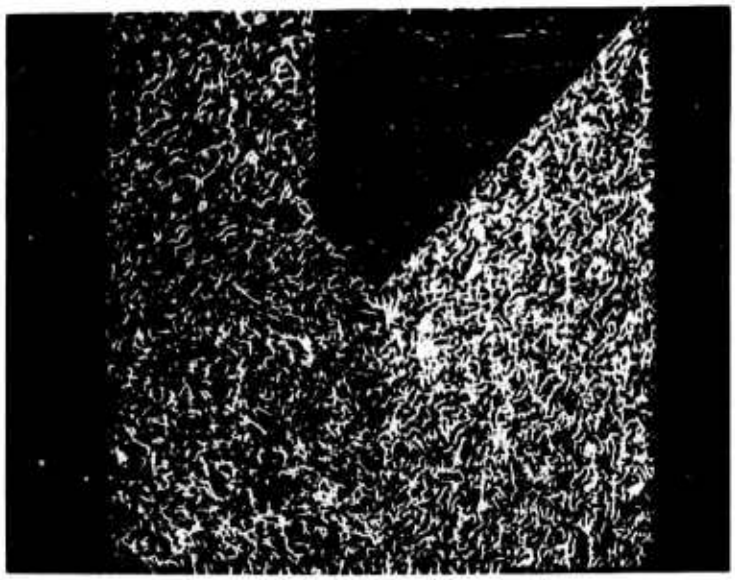


Figure 9

Extrema detected in Fig. 8

Figure 10

Edge values for Fig. 8 using a 64 x 48 window.

Figure 11

Texture edges detected in Fig. 8.

Figure 12

A more complex texture mosaic. It consists of 64 texture regions each 64 x 64.

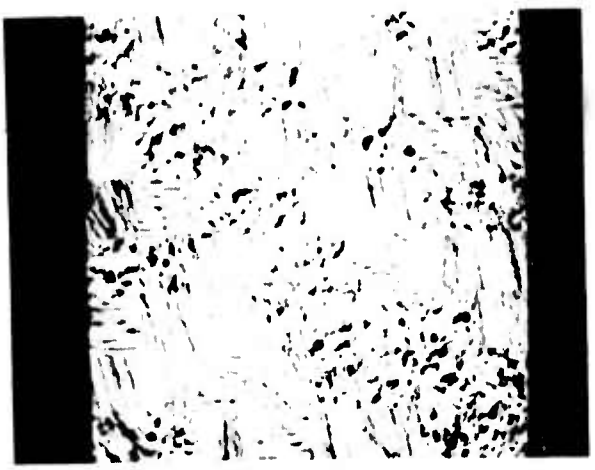


Figure 13

Texture edges detected in Fig. 12.

Figure 14

A 256 x 256 x-ray image.

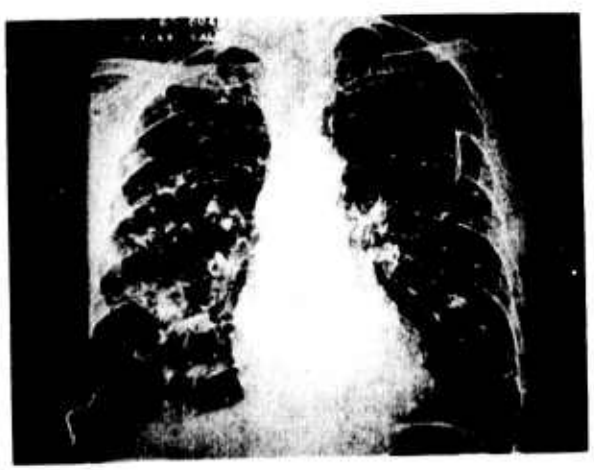


Figure 15

Maximum extrema detected in Fig. 14

Figure 16

Lowest two levels of maximum extrema detected in Fig. 14.

Figure 17

Edge values for Fig. 14 using a 16 x 16 window.

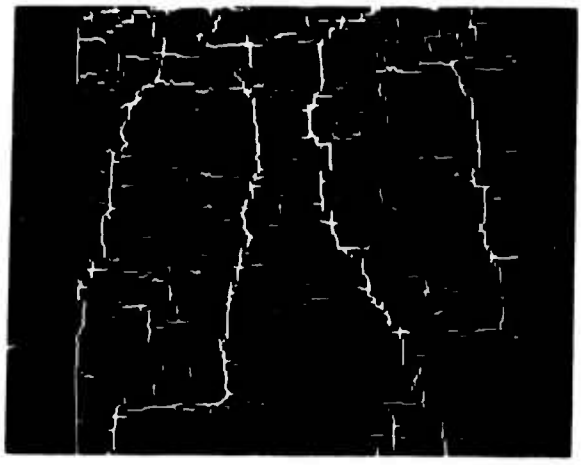


Figure 18

Texture edges detected in Fig. 14.

Figure 19

A 256 x 256 aerial image.

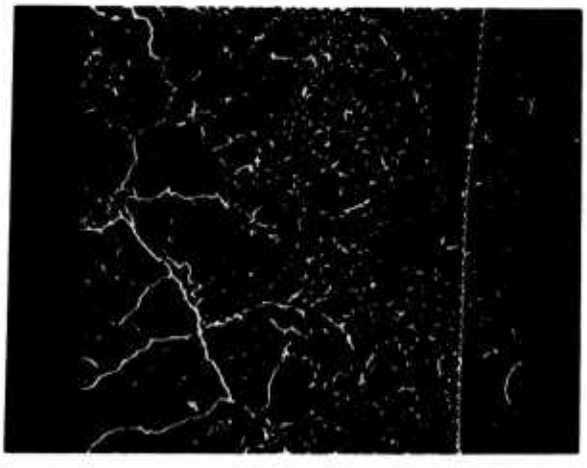


Figure 20

Maximum extrema detected in Fig. 19.

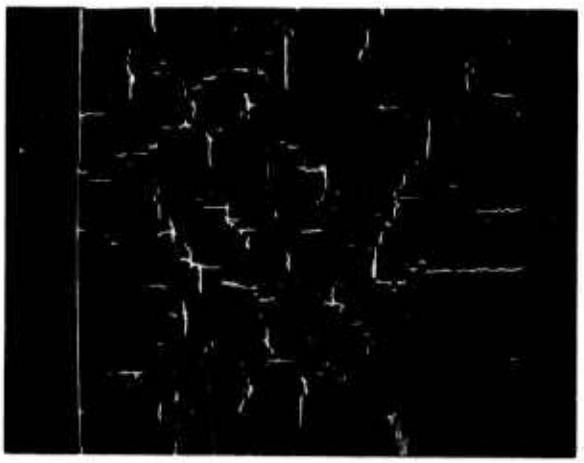


Figure 21

Texture edges detected in Fig. 19.

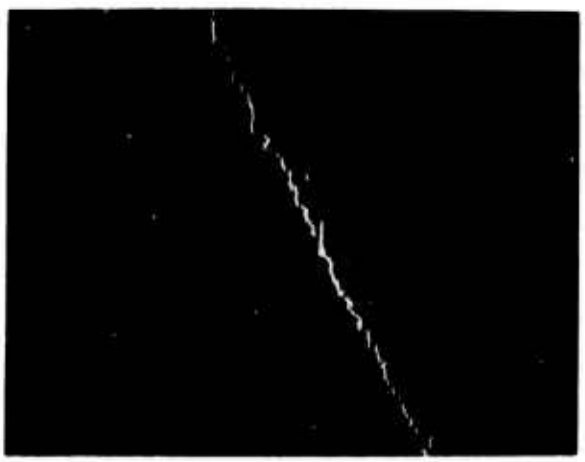


Plate 1

Superposition of texture regions (Fig. 1) and detected edge (Fig. 7).

Plate 2

Superposition of texture regions (Fig. 8) and detected edges (Fig. 11).

Plate 3

Superposition of texture regions (Fig. 12) and detected edges (Fig. 13).

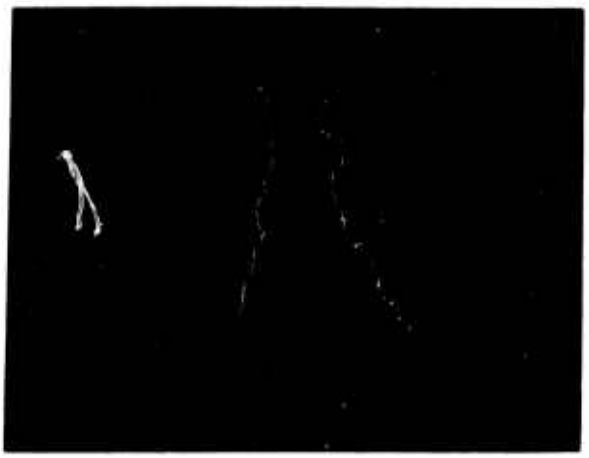


Plate 4

Superposition of x-ray (Fig. 14) and detected texture edges (Fig. 18).

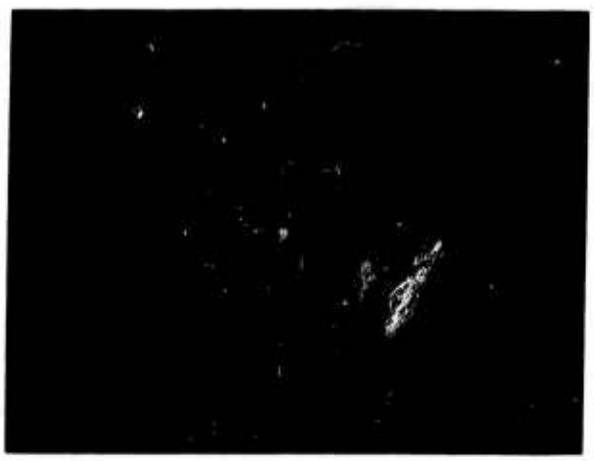


Plate 5

Superposition of aerial scene (Fig. 19) and detected texture edges (Fig. 20).

FOURIER SHAPE DESCRIPTORS

T. Wallace and P. A. Wintz

The Fourier descriptor (FD) is one method of describing the shape of a planar figure. Given a figure in the complex plane, the contour can be traced, yielding a complex function of time. If the contour is traced repeatedly, the periodic function which results can be expressed in a Fourier series. Granlund [1] defines the FD of a contour as the coefficients of this Fourier series.

To implement this method of shape description, it is necessary to sample the contour at a finite number of points. Since the discrete Fourier transform of a sequence gives us the values of the Fourier series coefficients of the sequence, assuming it to be periodic, using an FFT algorithm satisfies the definition above. The computational advantages of the FFT are well known.

The goal of this work is to classify the shapes of objects using their Fourier descriptors. The operations of rotation, scaling, and moving the starting point are easily implemented in the frequency domain by simple arithmetic on the frequency domain coefficients. While shapes may be compared in the space domain, the procedures required to adjust their size and orientation are computationally very expensive. Normally an iterative type of algorithm is employed, which searches for an optimum match between the unknown shape and the test set.

The goal of classification using Fourier descriptors is to develop an algorithm which will normalize the size and orientation of a shape before any comparisons to test shapes are made. If this can be accomplished, the classification process becomes a simple clustering problem with no iterative searches to contend with.

Let the output vector of the FFT be denoted (A(0),A(1),...,A(N)). The normalization procedure is as follows.

For position normalization simply set A(0) equal to some constant, or disregard it entirely, since it contains no information other than the position of the contour relative to the origin of the complex plane.

For magnitude normalization, multiply all the coefficients by a factor chosen such that the magnitude of some reference coefficient is a specified value. For most contours traced in the counterclockwise direction, A(1) will be the largest coefficient, so the obvious procedure is to set MAG (A(1)) equal to unity.

The other shape-preserving operations which must be performed in the normalization procedure affect orientation and starting point. To rotate the contour, all coefficients may be multiplied by EXP (Jθ). Where θ is the angle of rotation. To move the starting point from which the contour is traced, multiply the Ith coefficient by EXP (JIT), where T is the distance along the contour which the starting point is to be moved (the length of the contour is defined to be 2PI).

If the unknown shape is an exact match to a test shape, normalizing the orientation and starting point is straightforward. Since there are two permissible operation affecting the phase of the coefficients, the reference orientation and starting point can be defined by specifying the phase of two selected coefficients, such as A(1) and A(2).

When the unknown shape is not a match to any of the test shapes, and we would like to determine which of the test shapes it is closest to, the problem becomes more difficult. The same procedure can be followed, but if the difference between two similar shapes lies mainly in the particular coefficients used in normalization, an optimum orientation and starting point may not be

achieved. A simple application of Parseval's theorem shows that the square of the Euclidean distance in the frequency domain corresponds to a point by point mean square error in the space domain. It is evident that a small error in rotation or starting point normalization may cause the whole normalization procedure to break down.

It can be easily shown that in the case of contours with bilateral symmetry, if the starting point is chosen to lie on the axis of symmetry, the phases of all the coefficients will be equal to zero, or 180 degrees. This indicates that there is less information contained in the phases of the FD than in the magnitudes, so a classification using the magnitudes alone should be effective. If the contours to be classified are perfectly bilaterally symmetric, there is no problem in normalizing the FDs, since the procedure above will work perfectly. However, in practice, this is highly unlikely, due to noise and quantization error.

A set of aircraft silhouettes was used to test the magnitude classification method for bilaterally symmetric contours. Eighteen planes were digitized, the contours traced, and the FDs computed. Figures 1 thorugh 18 show the contours used in the experiment. Figure 19 shows the Euclidean distances in the frequency domain between the magnitudes of the FDs. Planes 1 through 9 are rear engine types, planes 10 through 12 are two engine types, and planes 13 through 18 are four engine types. The original contours contained 128 points, but only the first 48 frequency domain components were used in the classification; the higher coefficients contain information about quantization and very small detail, and have little effect on the distances.

If the plane being classified is deleted from the test set, and its type is assumed to be that of the nearest plane in the set, only one error occurs. Plane 11 is slightly nearer to Plane 15 than to Plane 12. It should be noted

however, that there are only three planes in the two engine category, so we probably do not have a very complete test set. It should also be noted that classifying purely on engine location is not synonymous with matching similar shapes; factors such as wing sweep and length to wingspan ratio are neglected. The influence of some of these latter factors is evident from an examination of the original contours. Plane 1, for example, while not misclassified, has a much smaller length to wingspan ratio than the other rear engine planes. This ratio is near unity, which is similar to most of the four engine planes. This explains why Plane 1 is not very close to any of the other planes, comparatively speaking. It also explains why Plane 1 is closer to the four engine group than is any other two engine model.

REFERENCES

1. G. H. Granlund, "Fourier Preprocessing for Hand Print Character Recognition," IEEE Trans. on Computers, vol. C-21, pp. 195-201, Feb. 1972.

Figure 1

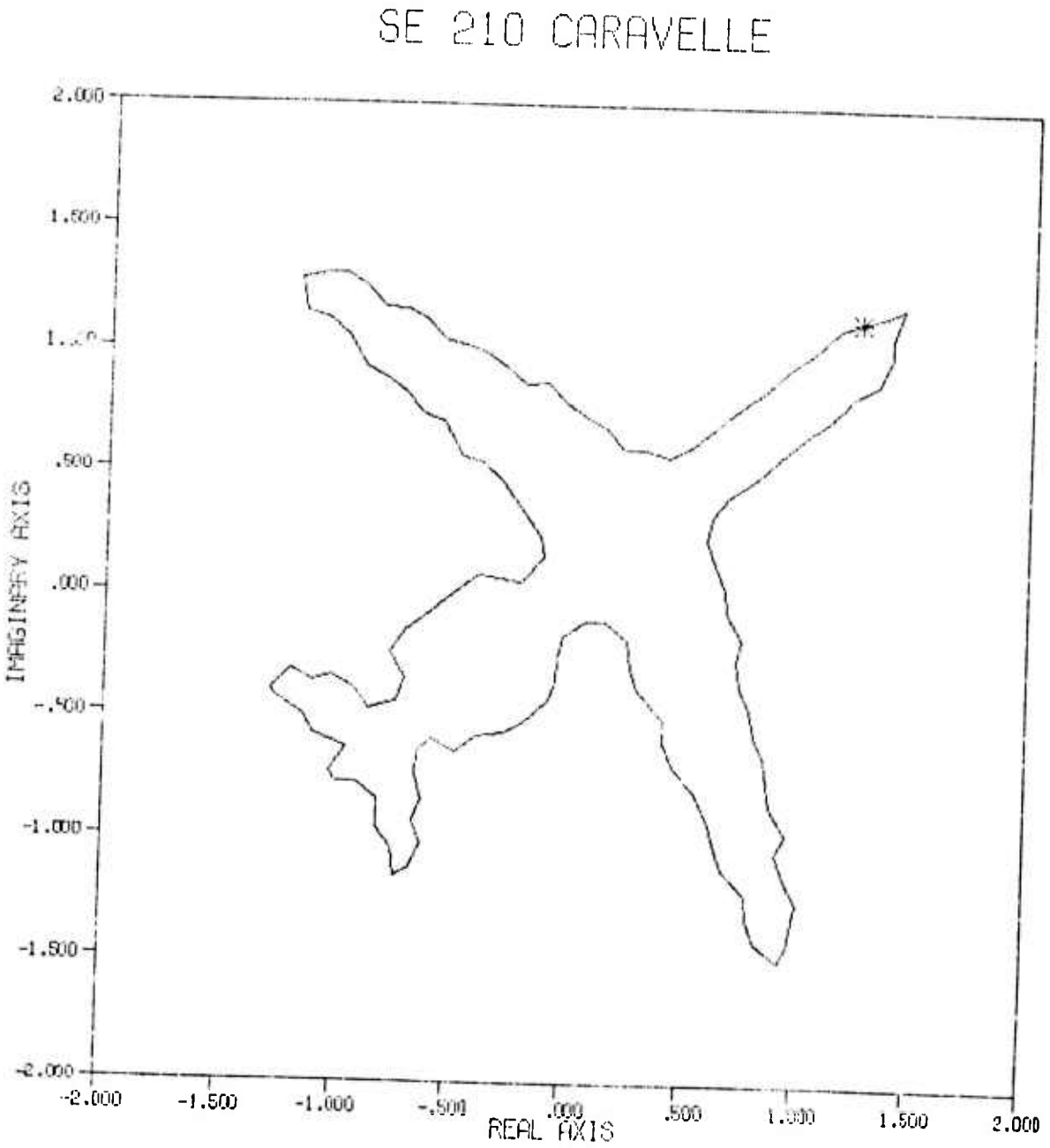

SE 210 CARAVELLE
2.000
1.500
.500
.000
-.500
-1.000
-1.500
-2.000
IMAGINARY AXIS
-2.000
-1.500
-1.000
-.500
.000
.500
1.000
1.500
2.000
REAL AXIS

Figure 2

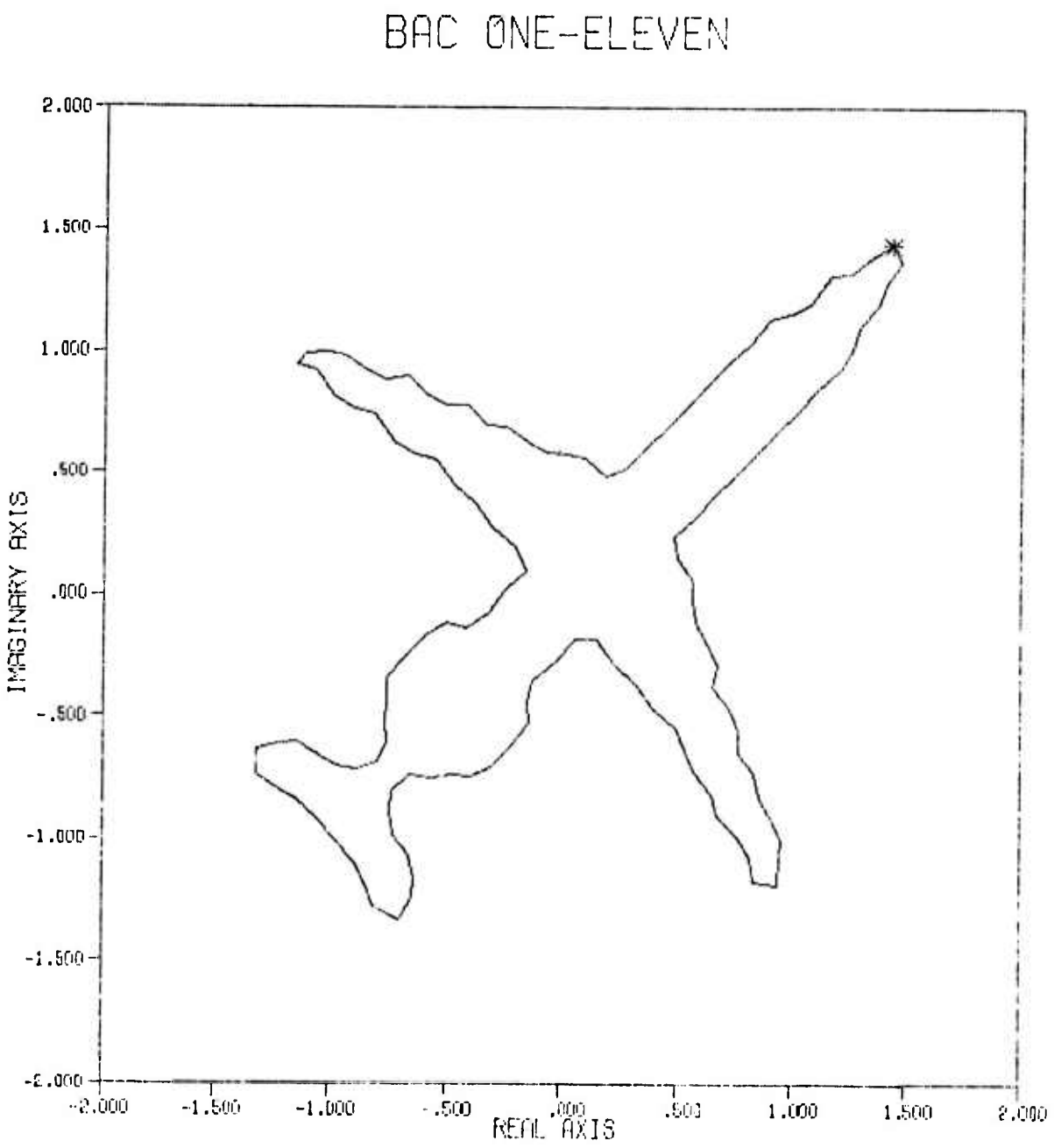

BAC ONE-ELEVEN
2.000
1.500
1.000
.500
.000
-.500
-1.000
-1.500
-2.000
IMAGINARY AXIS
-2.000
-1.500
-1.000
-.500
.000
.500
1.000
1.500
2.000
REAL AXIS

Figure 3

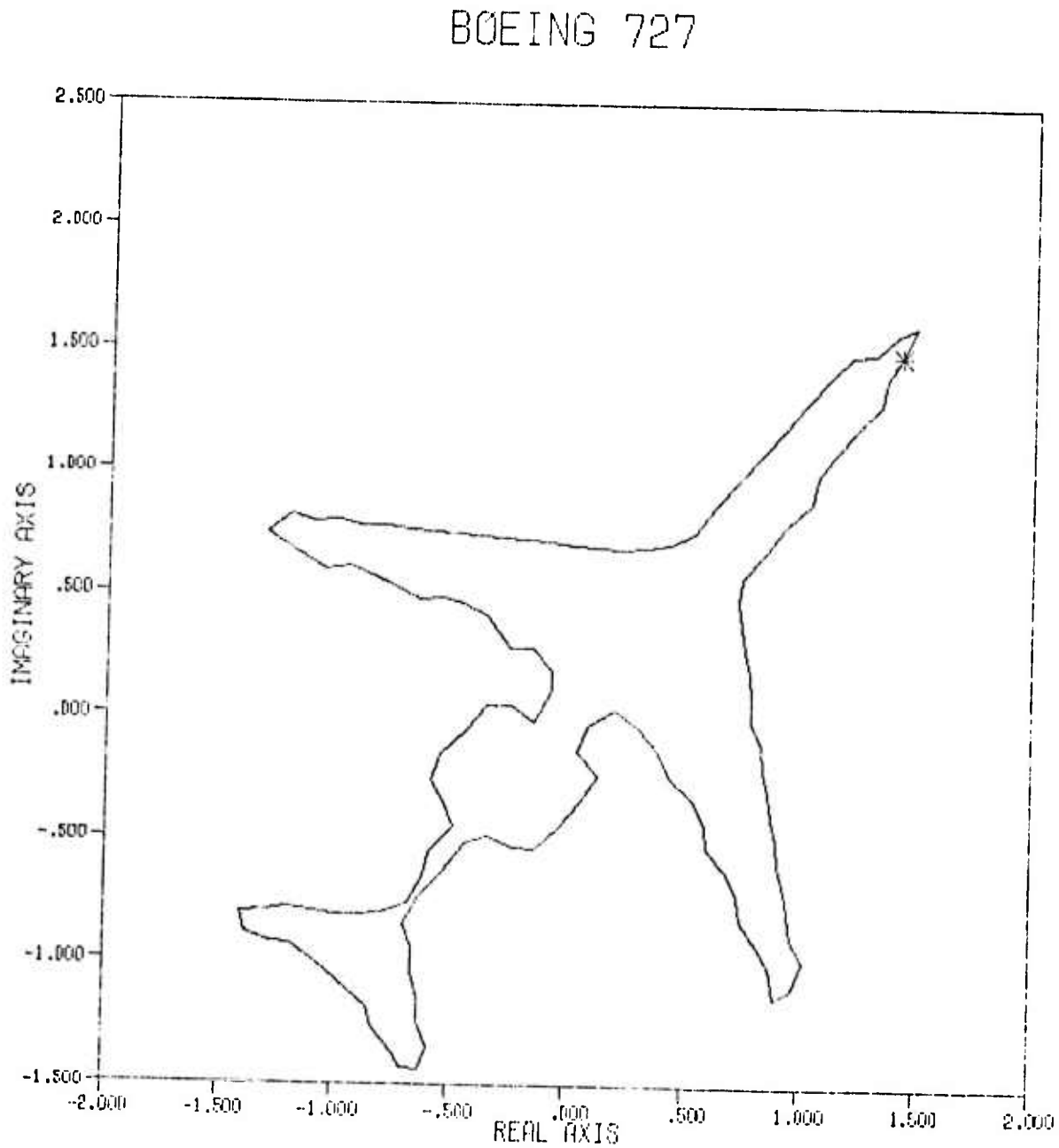

BOEING 727
2.500
2.000
1.500
1.000
.500
.000
-.500
-1.000
-1.500
IMAGINARY AXIS
-2.000
-1.500
-1.000
-.500
.000
.500
1.000
1.500
2.000
REAL AXIS

Figure 4

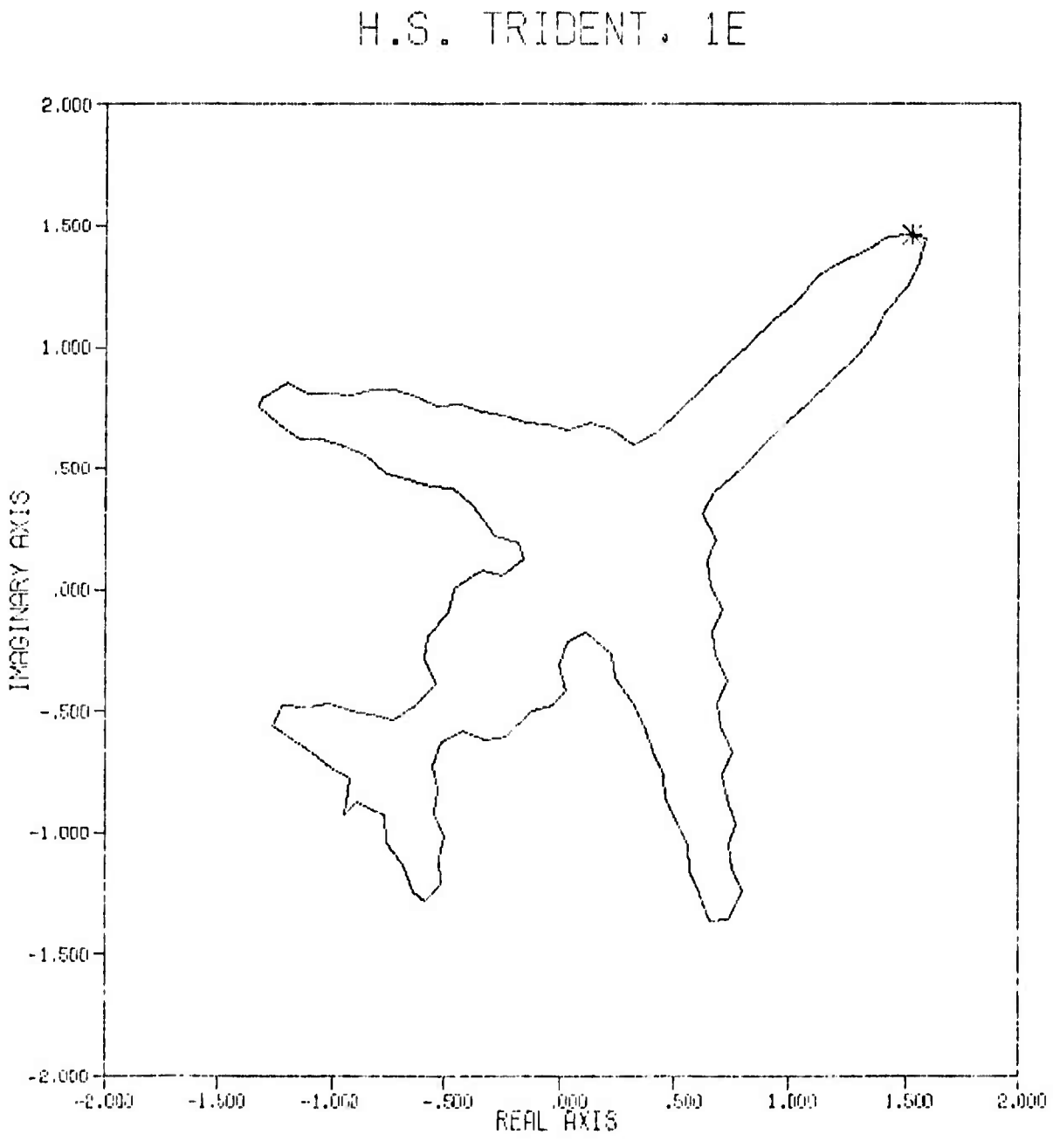

H.S. TRIDENT. 1E
2.000
1.500
1.000
.500
.000
-.500
-1.000
-1.500
-2.000
IMAGINARY AXIS
-2.000
-1.500
-1.000
-.500
.000
.500
1.000
1.500
2.000
REAL AXIS

Figure 5

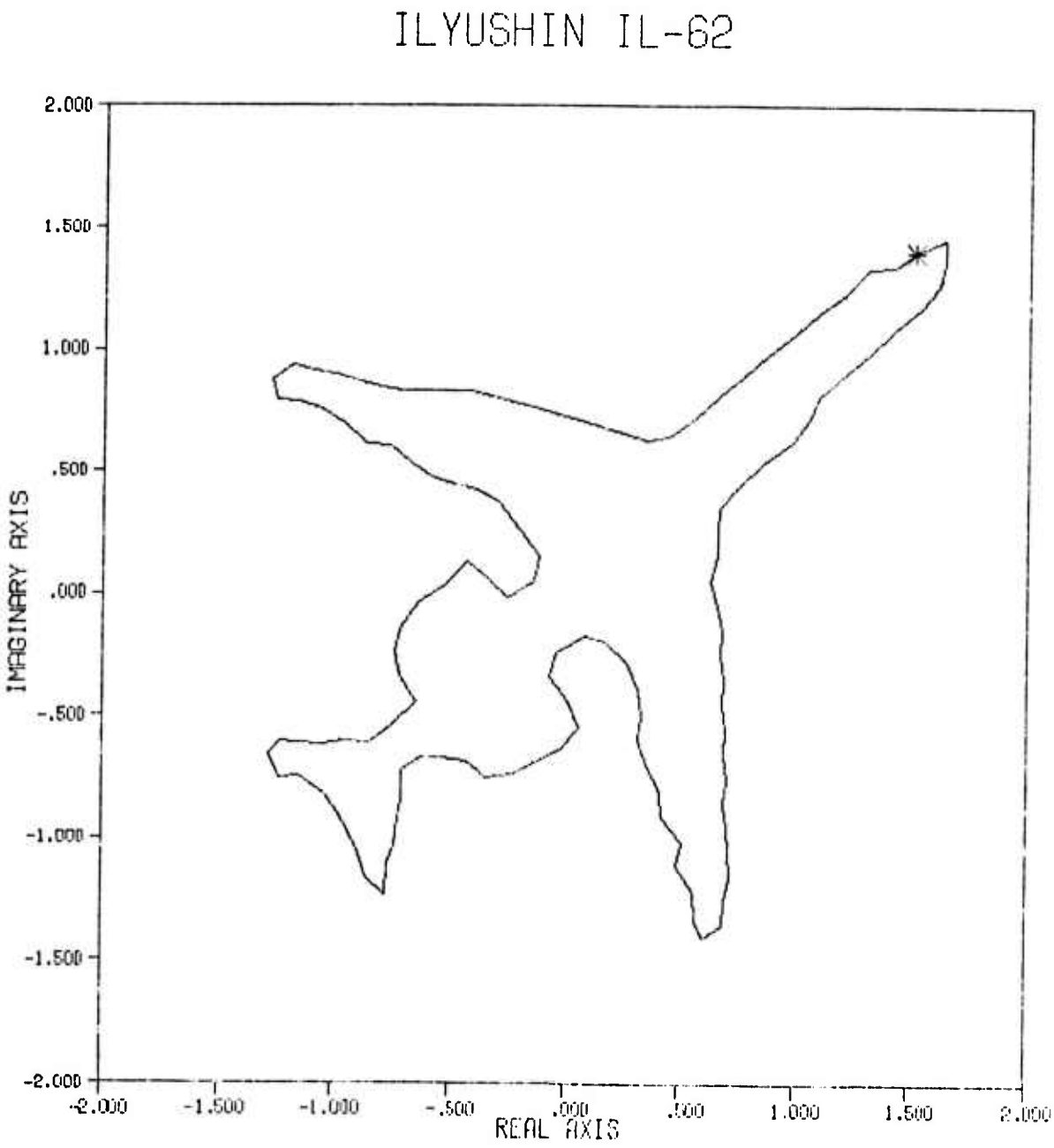

ILYUSHIN IL-62
2.000
1.500
1.000
.500
.000
-.500
-1.000
-1.500
-2.000
IMAGINARY AXIS
-2.000
-1.500
-1.000
-.500
.000
.500
1.000
1.500
2.000
REAL AXIS

Figure 6

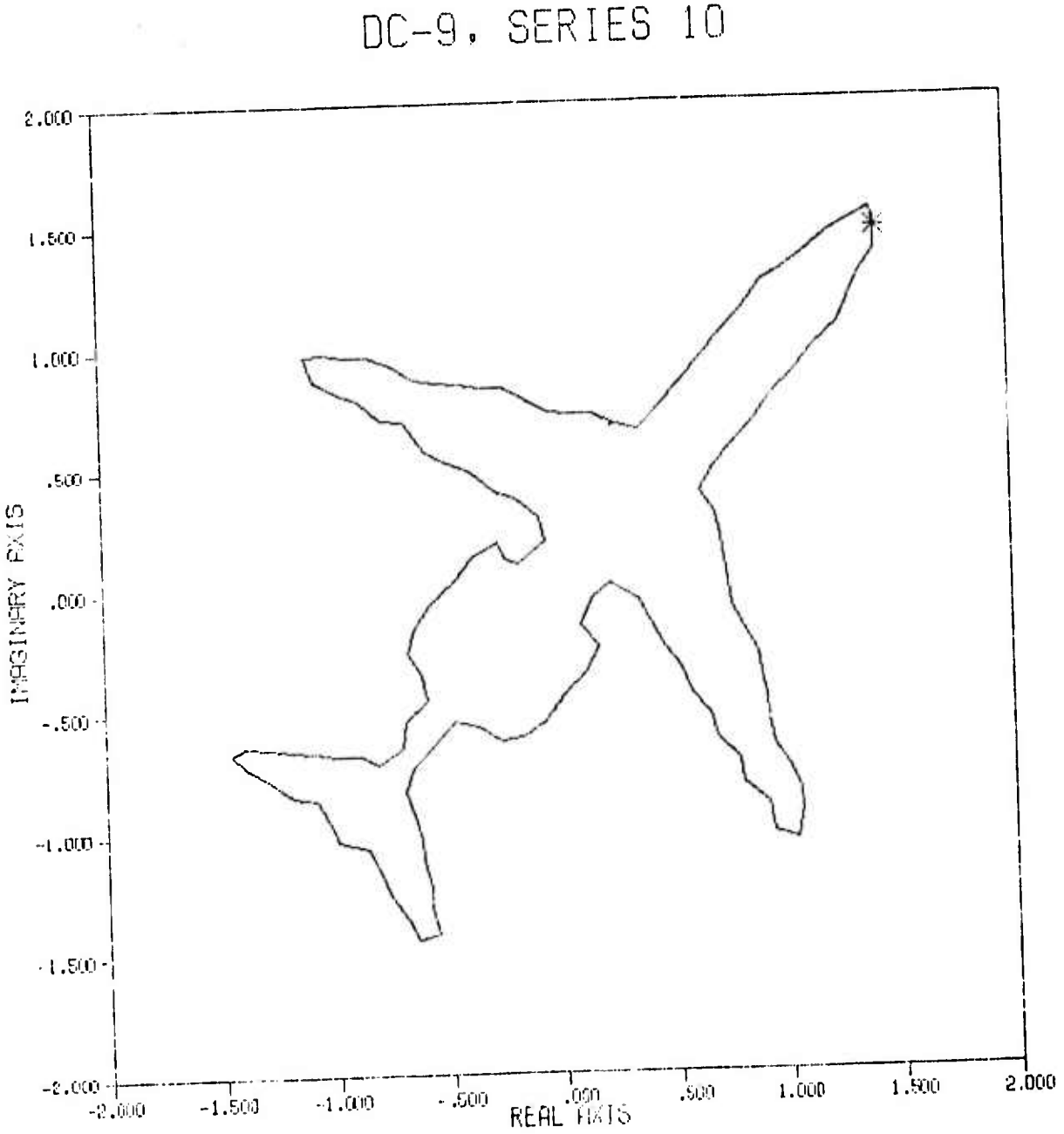

DC-9, SERIES 10
2.000
1.500
1.000
.500
.000
-.500
-1.000
-1.500
-2.000
IMAGINARY AXIS
-2.000
-1.500
-1.000
-.500
.000
.500
1.000
1.500
2.000
REAL AXIS

Figure 7

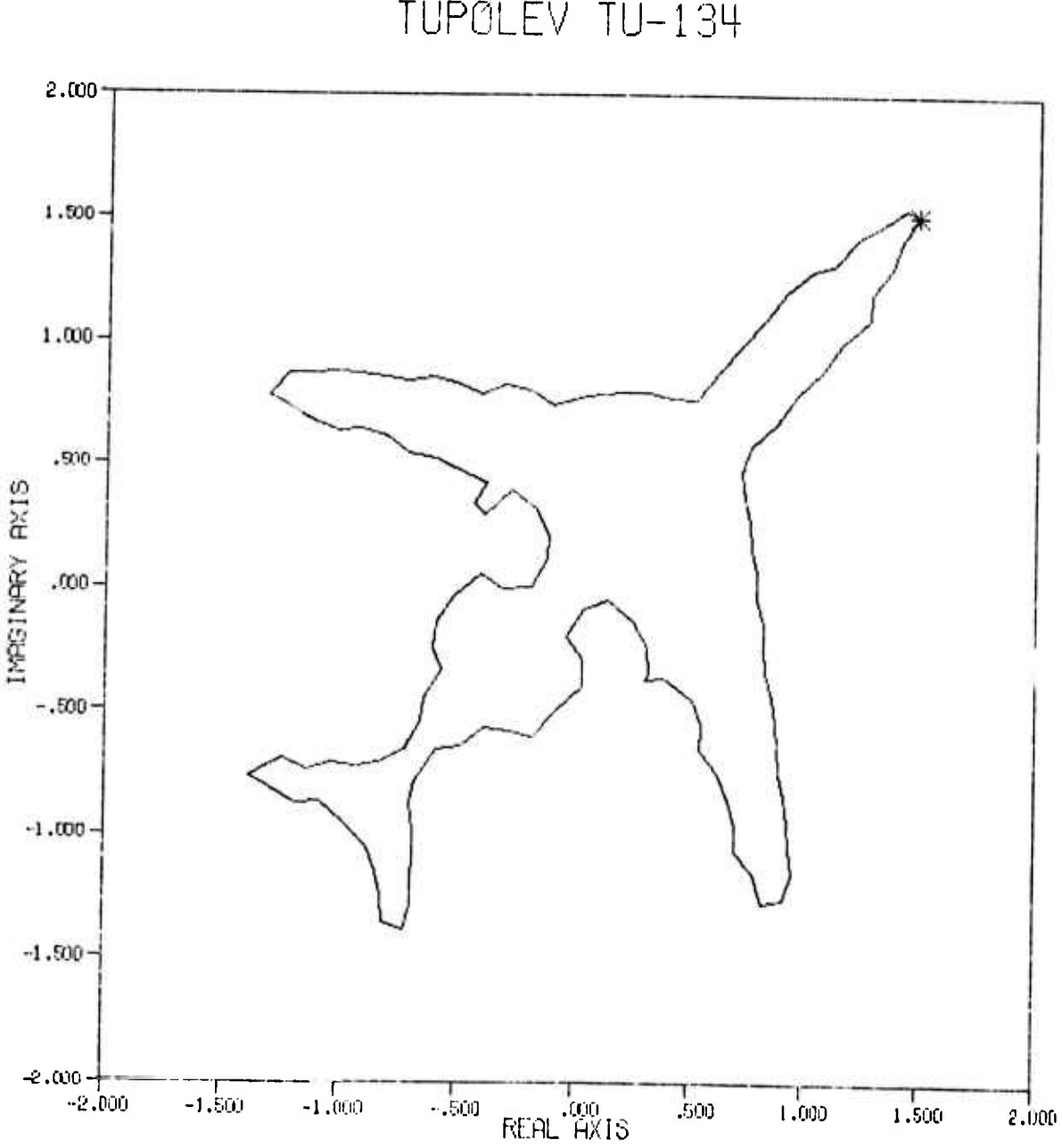


Figure 8

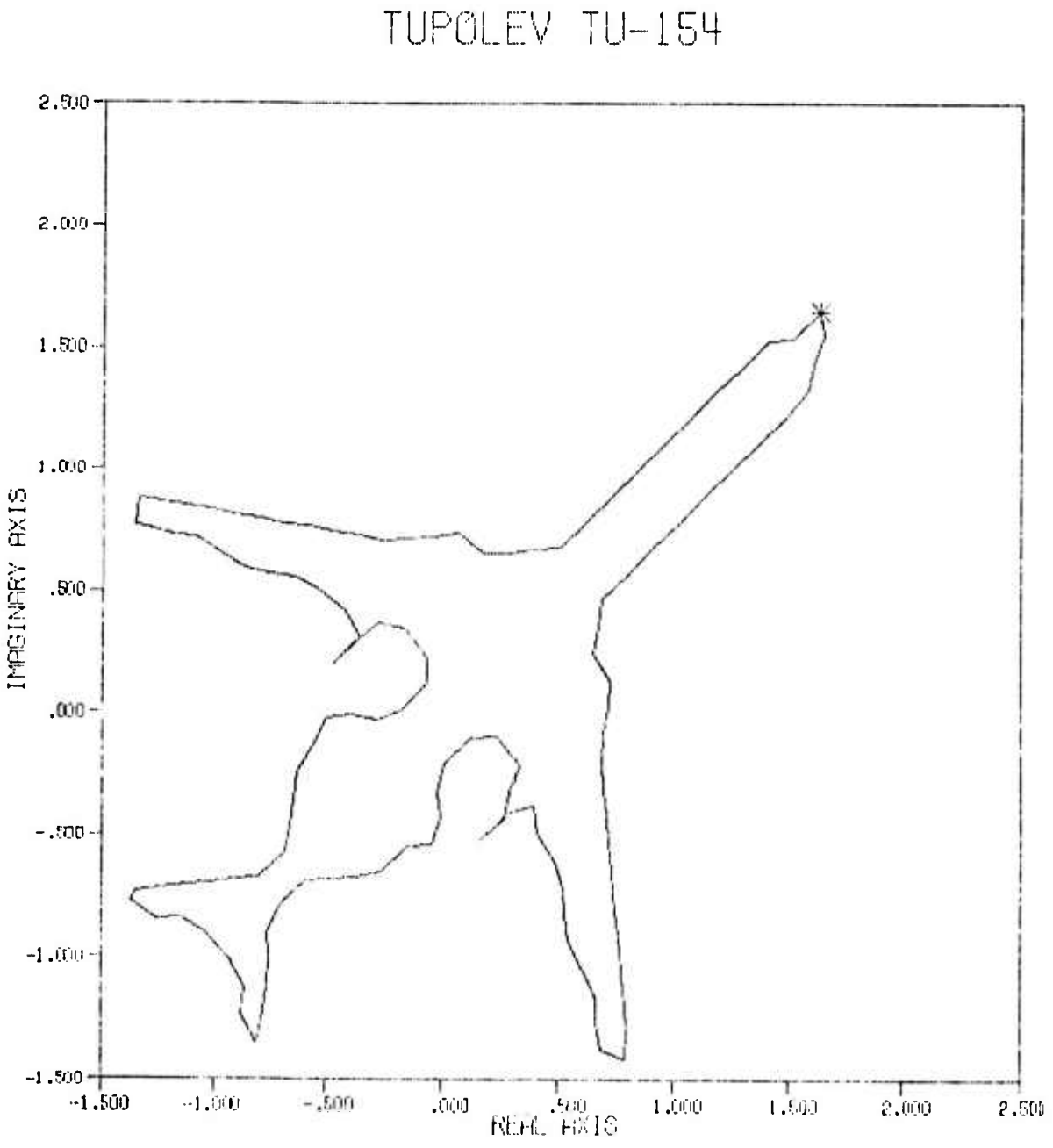

TUPOLEV TU-154
2.500
2.000
1.500
1.000
.500
.000
-.500
-1.000
-1.500
IMAGINARY AXIS
-1.500
-1.000
-.500
.000
.500
1.000
1.500
2.000
2.500
REAL AXIS

Figure 9

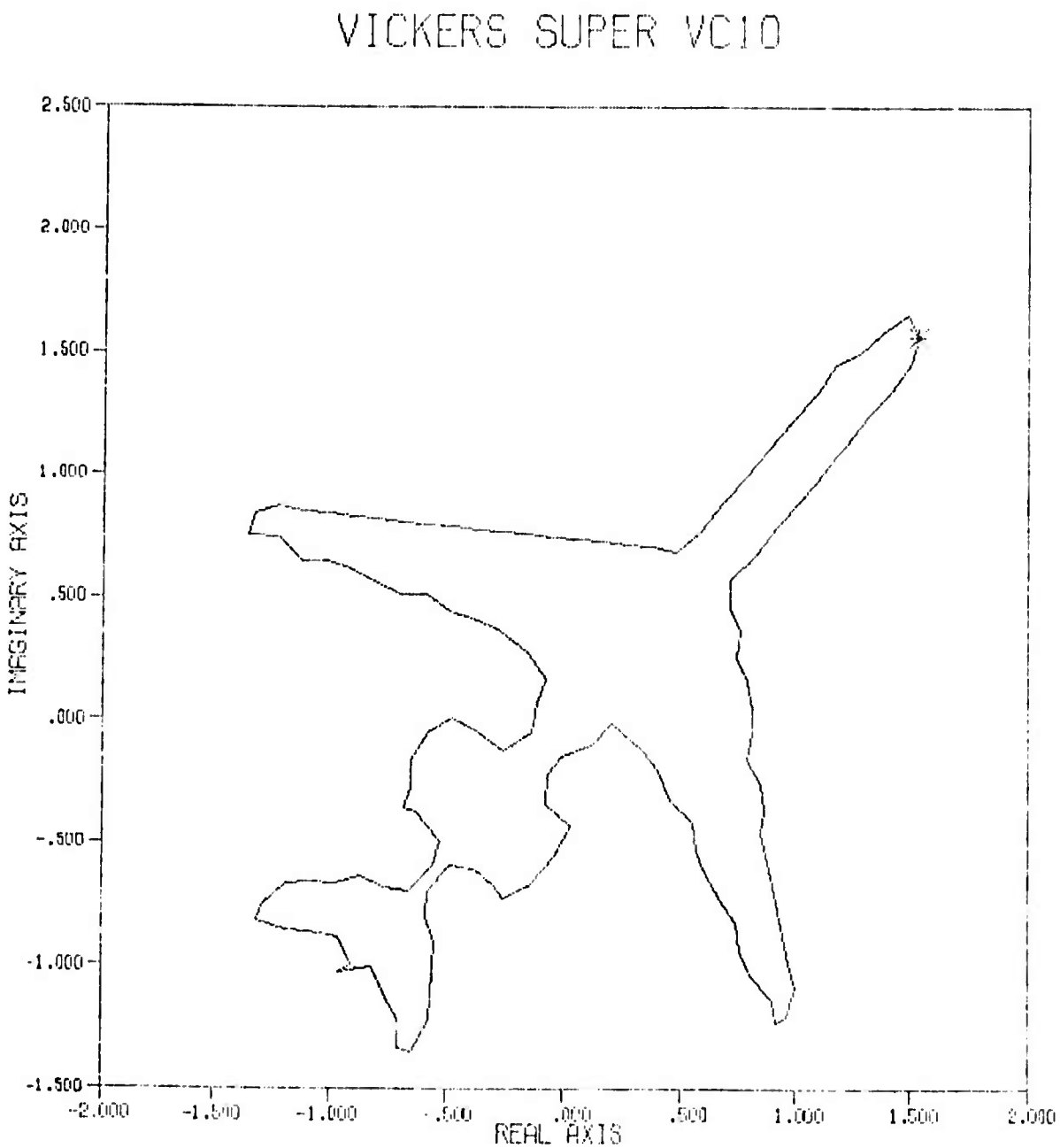

VICKERS SUPER VC10
2.500
2.000
1.500
1.000
.500
.000
-.500
-1.000
-1.500
IMAGINARY AXIS
-2.000
-1.500
-1.000
-.500
.000
.500
1.000
1.500
2.000
REAL AXIS

Figure 10

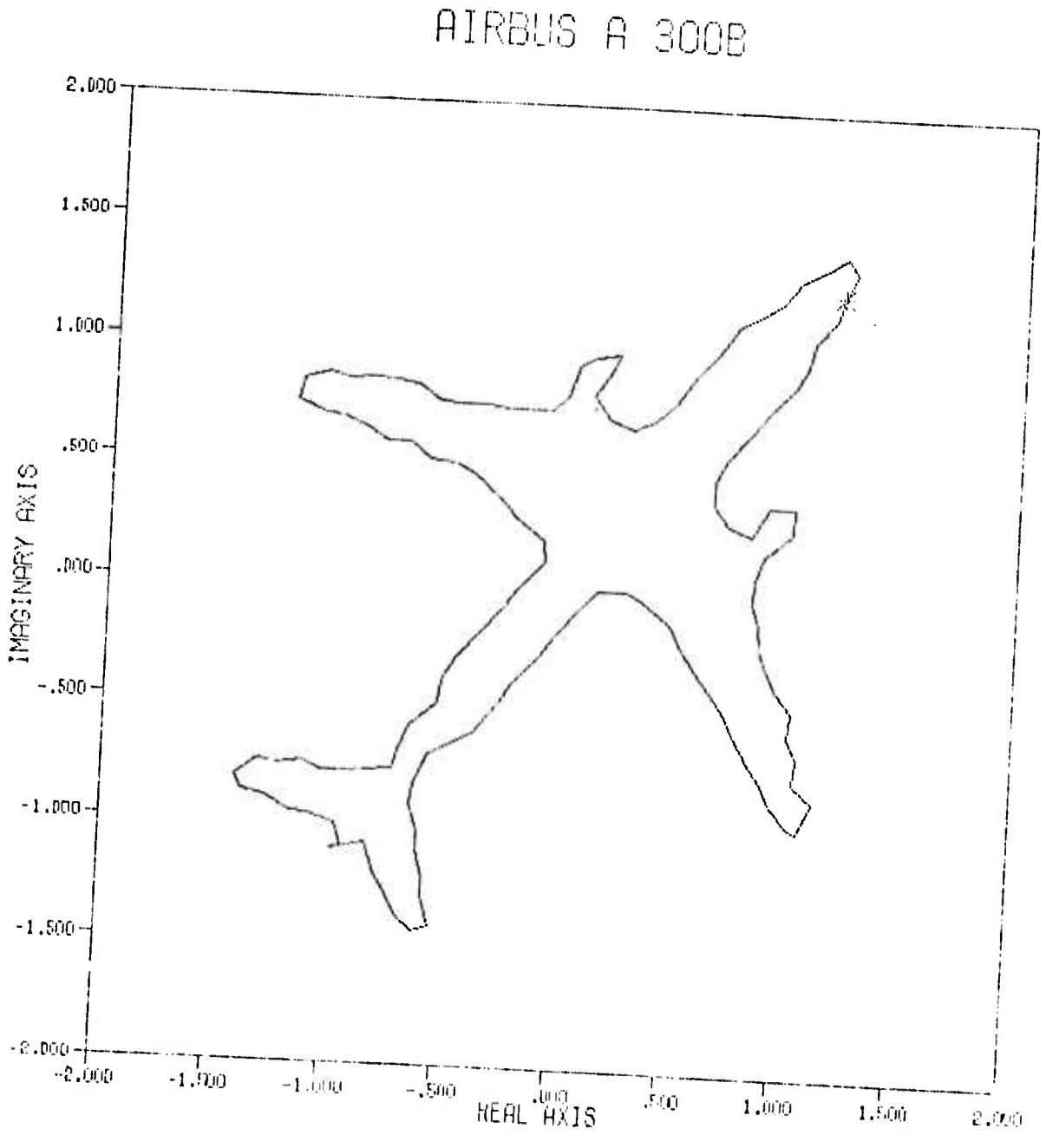

AIRBUS A 300B
2.000
1.500
1.000
.500
.000
-.500
-1.000
-1.500
-2.000
IMAGINARY AXIS
-2.000
-1.500
-1.000
-.500
.000
.500
1.000
1.500
2.000
REAL AXIS

Figure 11

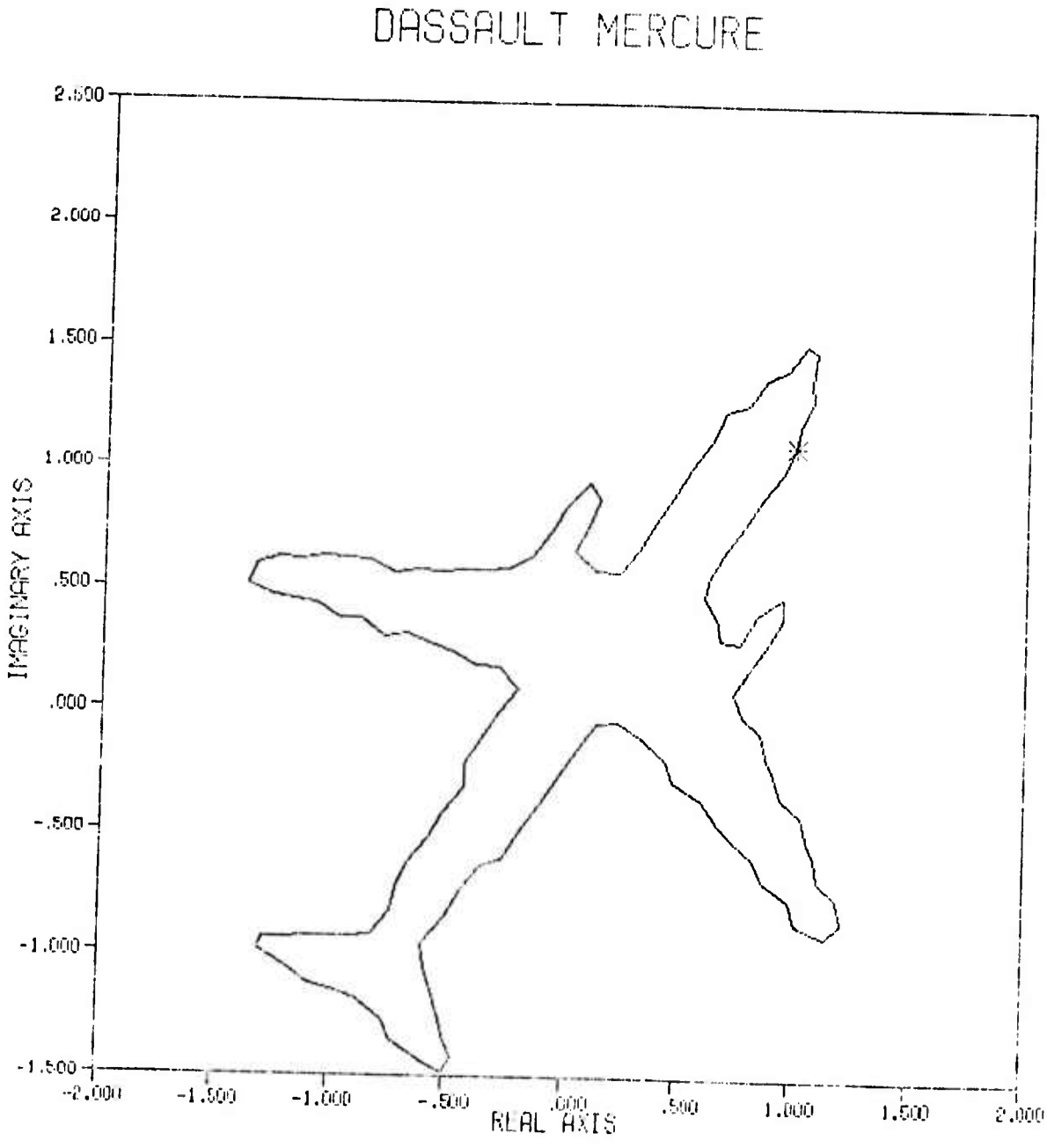

DASSAULT MERCURE
2.500
2.000
1.500
1.000
.500
.000
-.500
-1.000
-1.500
IMAGINARY AXIS
-2.000
-1.500
-1.000
-.500
.000
.500
1.000
1.500
2.000
REAL AXIS

Figure 12

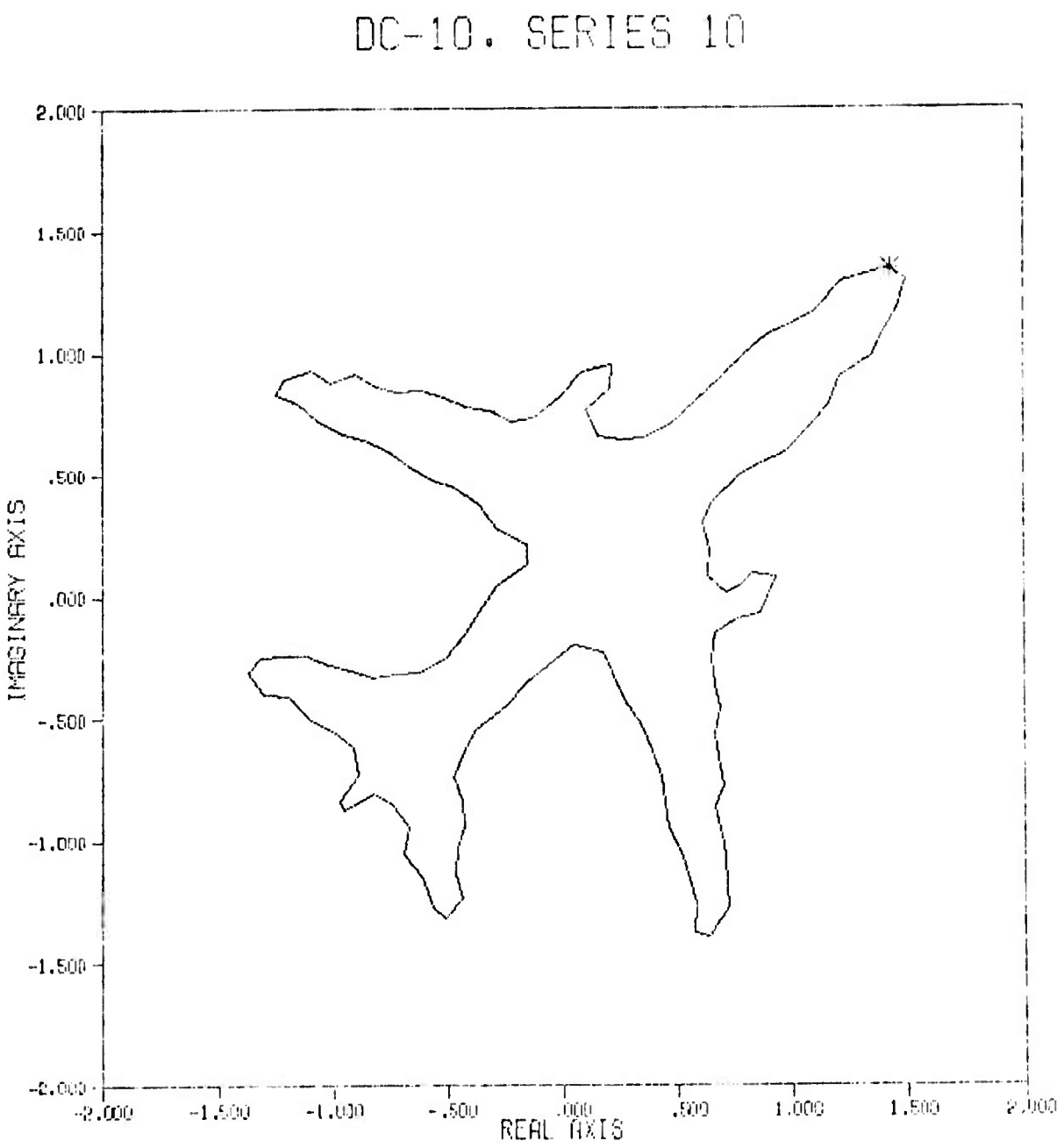

DC-10. SERIES 10
2.000
1.500
1.000
.500
.000
-.500
-1.000
-1.500
-2.000
IMAGINARY AXIS
-2.000
-1.500
-1.000
-.500
.000
.500
1.000
1.500
2.000
REAL AXIS

Figure 13

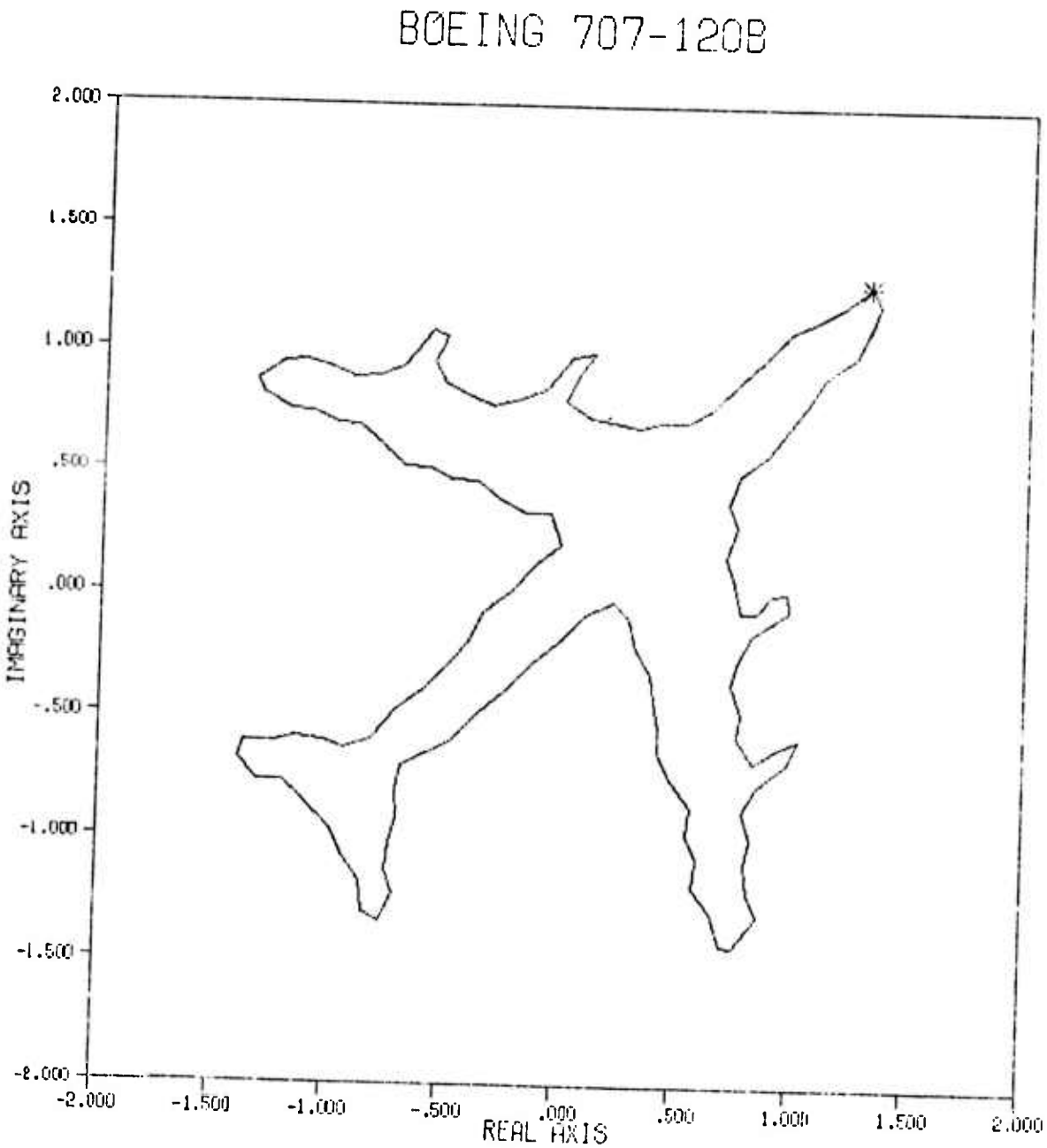

BOEING 707-120B
2.000
1.500
1.000
.500
.000
-.500
-1.000
-1.500
-2.000
IMAGINARY AXIS
-2.000
-1.500
-1.000
-.500
.000
.500
1.000
1.500
2.000
REAL AXIS

Figure 14

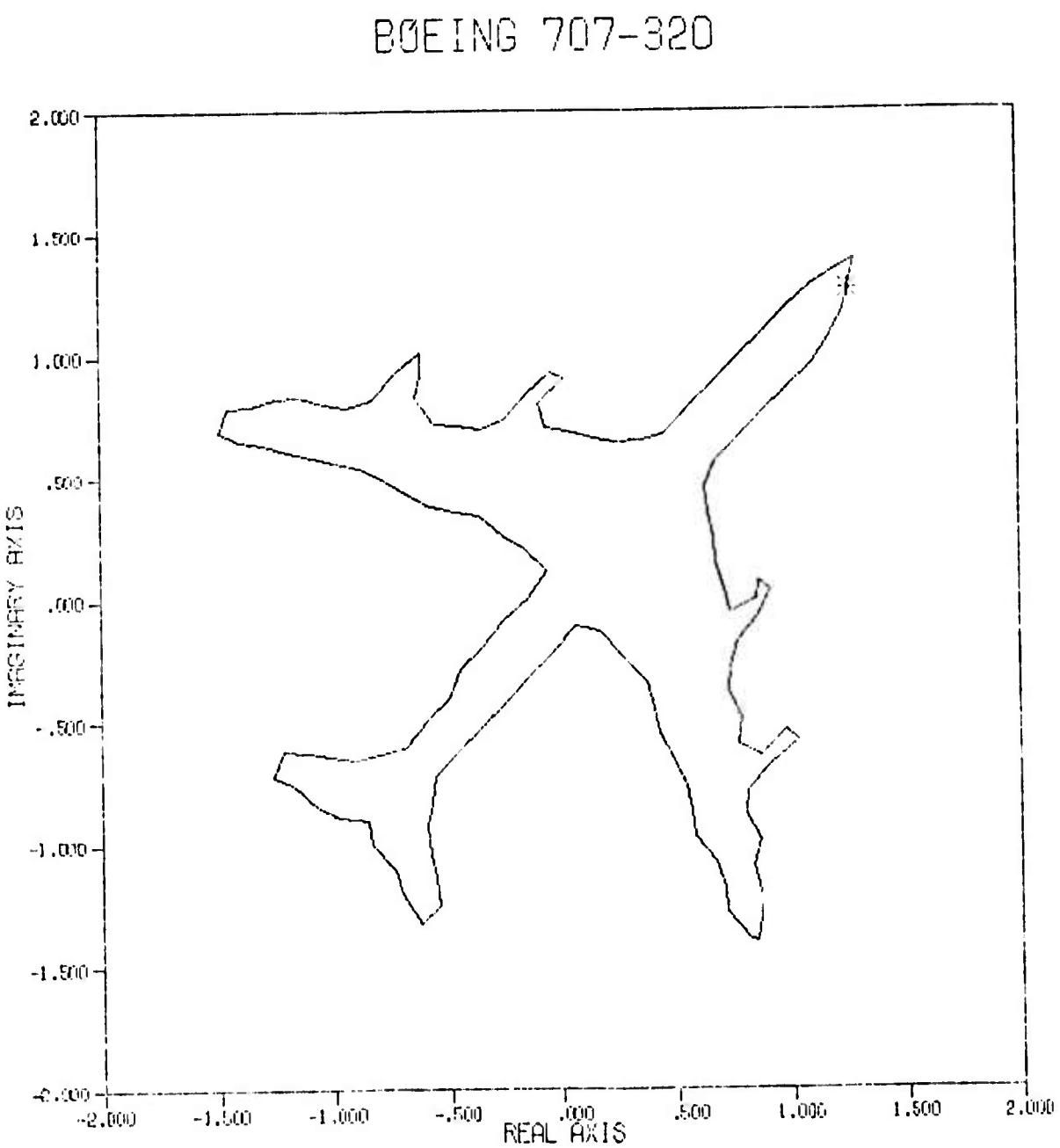

BØEING 707-320
2.000
1.500
1.000
.500
.000
-.500
-1.000
-1.500
-2.000
IMAGINARY AXIS
-2.000
-1.500
-1.000
-.500
.000
.500
1.000
1.500
2.000
REAL AXIS

Figure 15

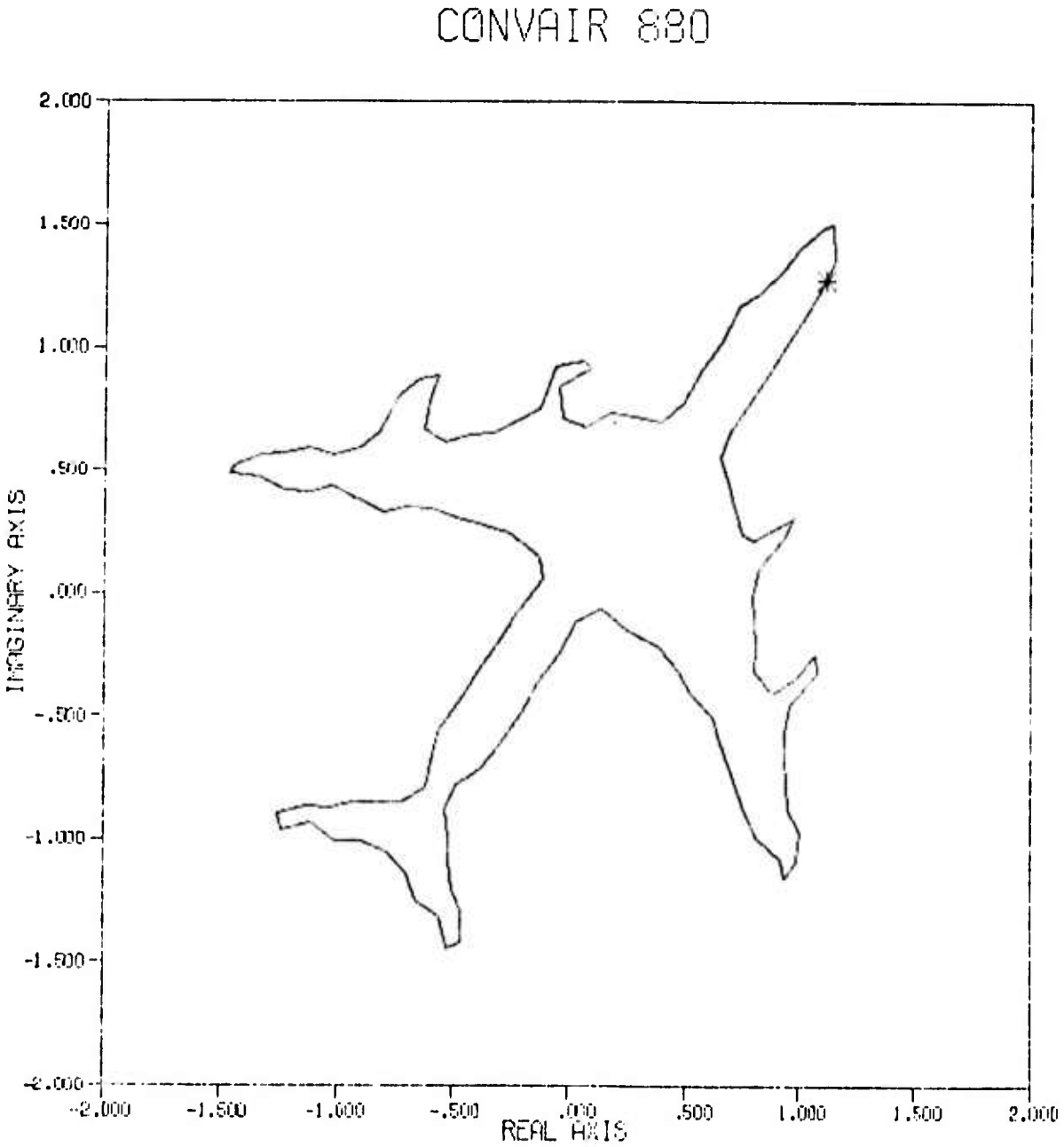

CONVAIR 880
2.000
1.500
1.000
.500
.000
-.500
-1.000
-1.500
-2.000
IMAGINARY AXIS
-2.000
-1.500
-1.000
-.500
.000
.500
1.000
1.500
2.000
REAL AXIS

Figure 16

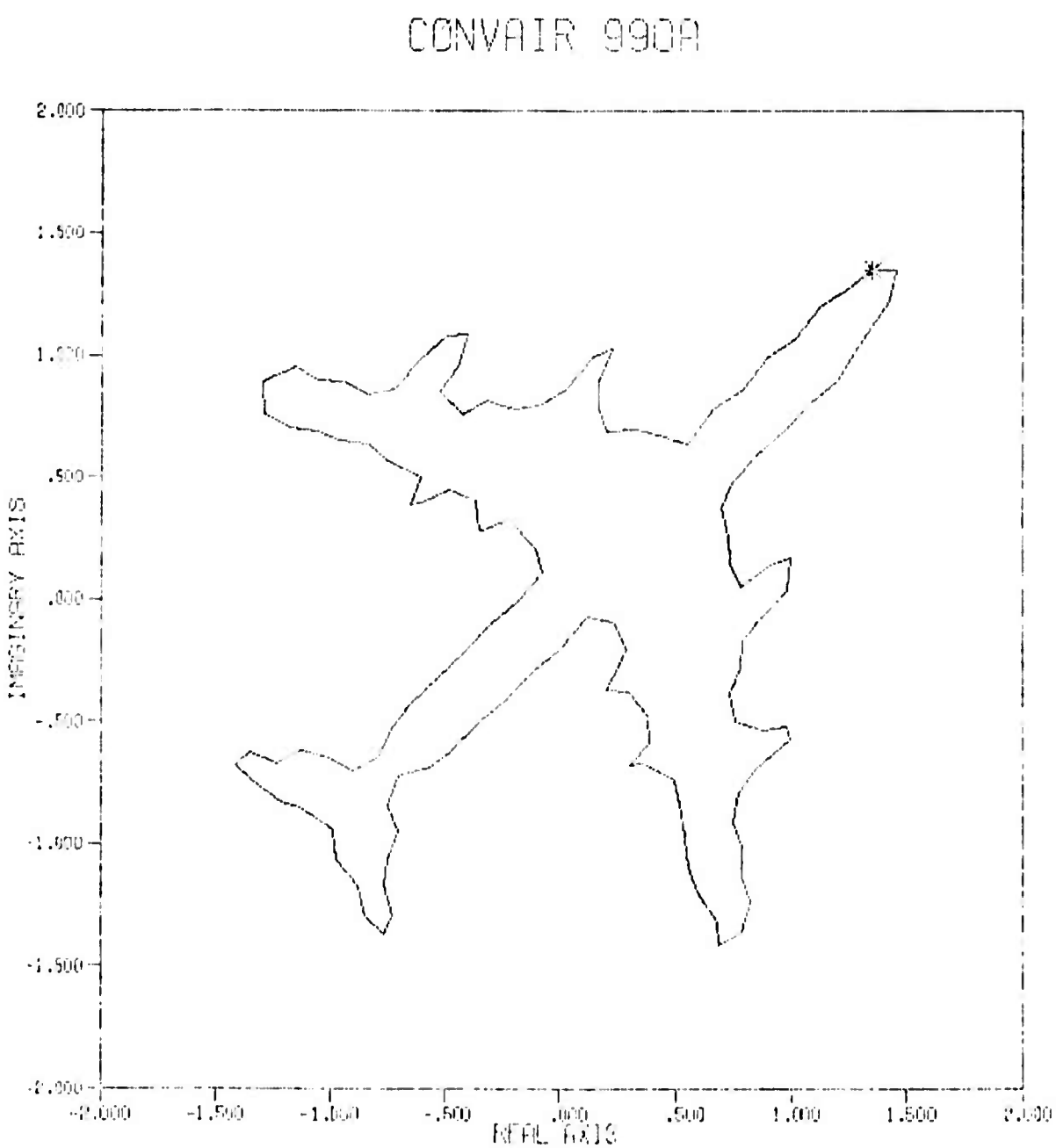

CONVAIR 990A
2.000
1.500
1.000
.500
.000
-.500
-1.000
-1.500
-2.000
IMAGINARY AXIS
-2.000
-1.500
-1.000
-.500
.000
.500
1.000
1.500
2.000
REAL AXIS

Figure 17

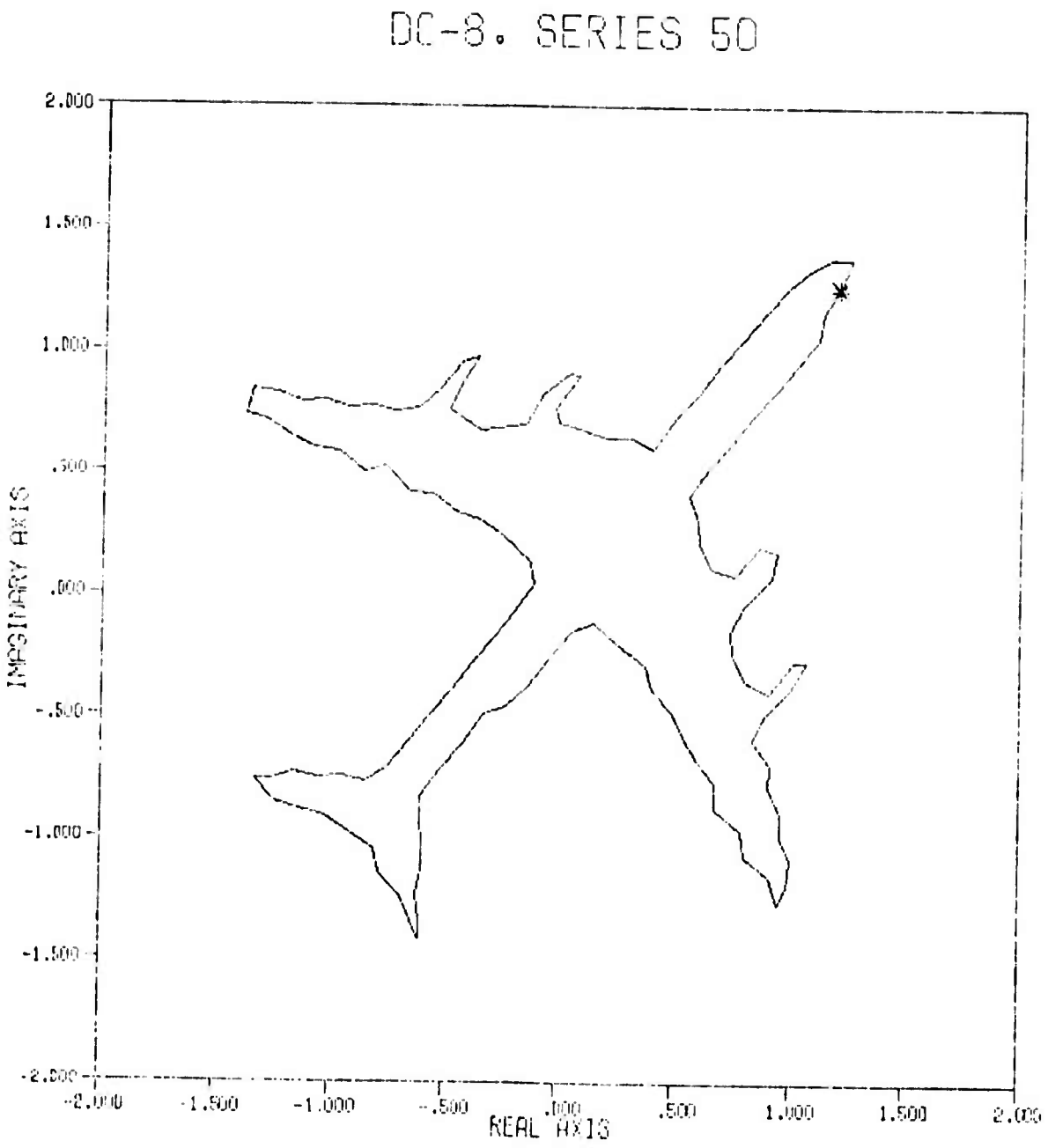
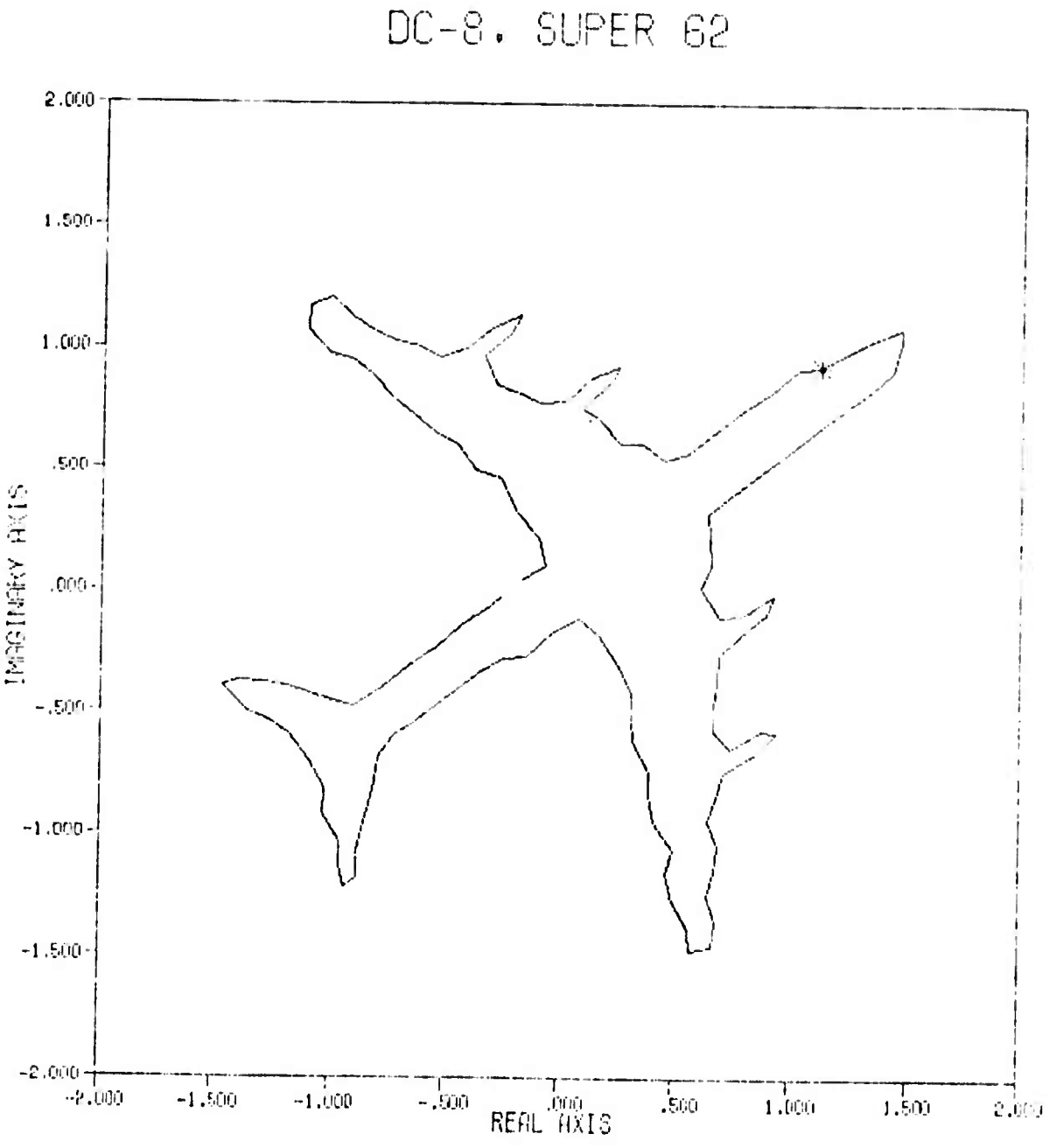

DC-8. SERIES 50
2.000
1.500
1.000
.500
.000
-.500
-1.000
-1.500
-2.000
IMPSINARY AXIS
-2.000
-1.500
-1.000
-.500
.000
.500
1.000
1.500
2.000
REAL AXIS

Figure 18


DC-8. SUPER 62
2.000
1.500
1.000
.500
.000
-.500
-1.000
-1.500
-2.000
IMAGINARY AXIS
-2.000
-1.500
-1.000
-.500
.000
.500
1.000
1.500
2.000
REAL AXIS

DISTANCES BETWEEN NORMALIZED FOURIER DESCRIPTORS OF AIRPLANE CONTOURS

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	0	459	383	284	412	664	296	526	342	483	661	496	478	603	578	507	736	686
2	459	0	273	161	182	523	344	448	358	347	366	285	725	687	522	569	670	683
3	383	273	0	199	200	207	60	241	69	370	684	547	901	1051	817	664	1055	1033
4	284	161	199	0	92	588	165	229	241	423	426	258	511	568	455	363	628	609
5	412	182	200	92	0	581	147	139	189	556	576	412	703	749	617	463	834	816
6	664	523	207	588	581	0	322	673	235	444	1044	827	1463	1577	1276	1200	1542	1541
7	296	344	60	165	147	322	0	131	74	474	714	551	766	913	766	538	996	964
8	526	448	241	229	139	673	131	0	222	681	914	760	854	938	888	582	1119	1097
9	342	358	69	241	189	235	74	222	0	540	914	665	1004	1111	952	771	1236	1207
10	483	347	370	423	556	444	474	881	540	0	292	281	795	935	597	681	749	717
11	661	366	684	428	576	1044	710	914	914	292	0	206	369	417	191	306	233	224
12	496	265	547	256	412	827	551	760	665	281	206	0	463	471	272	402	406	385
13	478	725	901	511	703	1463	766	854	1004	795	369	463	0	105	92	99	114	85
14	603	687	1051	568	749	1577	913	938	1111	935	417	471	105	0	104	227	115	110
15	578	522	817	455	617	1276	766	888	952	597	191	272	92	104	0	142	38	31
16	507	569	664	363	463	1200	538	582	771	681	306	402	99	227	142	0	210	189
17	736	670	1055	628	834	1542	996	1119	1236	749	233	406	114	115	38	210	0	8
18	686	683	1033	609	816	1541	964	1097	1207	717	224	385	85	110	31	189	8	0

Fig. 19

SYNTACTIC SCENE ANALYSIS

K.S. Fu, P.H. Swain, R.Y. Li, J. Keng, T.S. Yu

The bulk of the pattern recognition techniques employed to date for the machine analysis of aircraft or satellite gathered image data have concentrated on the spectral characteristics of the data, i.e., "color" in a generalized sense. Relatively little use has been made for image understanding purposes of the information contained in the spatial relationships in the data. The goal of this research is to characterize the spatial information content of such imagery in terms of (1) syntactic image models [1,2,3,4] and (2) statistical dependency models of context [5,6]. As a subtask, we shall also study change detection, a particularly significant practical aspect of spatial scene information.

(i) Syntactic image modeling.

The immediate thrust of this activity is to extend the work of Fu and Brayer [4], who developed grammars for describing image entities such as roads and cloud/shadow pairs. Linguistic rules in terms of tree or web grammars can be constructed which characterize such properties as shape, size, and texture as well as spatial relationships such as "surrounded by" or "near by". Directional references can also be explored to locate classes of large areas where no definite shapes exist, such as those found in land-use classification. A hierarchical or tree graph model can be built to contain the spatial distributions of all classes in the entire scene. We are investigating the extent to which syntactic rules in general, with tree structures in particular, can be used for generating shape descriptors for recognition purposes in the context of remote sensing data where traditional purely spectral approaches would not achieve satisfactory results.

At present, we have selected imagery of Lafayette, Indiana as a training data set. Out of seven primary ground cover types, we have chosen river and

highway because of their relatively homogeneous spectral characters and their linear shape patterns. A set of syntactic rules is to be inferred and subsequently applied to other data sets to test its generality.

(ii) Statistical dependency models of context

The traditional simple decision procedure is to select a state (i.e., classify a point as) belonging to the member of nature space $\Theta = (1,2,\ldots r)$ which will minimize the Bayes risk

$$\sum_{\theta=1}^{r} L(\theta,a)p(x|\theta)G(\theta) \tag{1}$$

where $L(\theta,a)$ is the loss function for $\theta\varepsilon\Theta$ and $a\varepsilon A = \Theta$. The loss function L is averaged over a prior distribution of θ. $G(\theta)$ is the prior probability of occurrence of class Θ. To introduce context into the model we use a compound decision rule. In this case we have $\underset{\sim}{\theta}_n = (\theta_1,\ldots\theta_n)$ as a set of states of nature and a set $\underset{\sim}{x}_n = (x_1,\ldots x_n)$ of vector-valued random variables to be classified jointly. We must choose a set of actions $\underset{\sim}{a}_n = (a_1,\ldots a_n)$ to minimize the expected loss for a compound loss function $L(\underset{\sim}{\theta}_n,\underset{\sim}{a}_n)$ where $L(\underset{\sim}{\theta}_n,\underset{\sim}{a}_n)$ is the average of the losses incurred in each of the n decisions.

$$L(\underset{\sim}{\theta}_n,\underset{\sim}{a}_n) = \frac{1}{n}\sum_{k=1}^{n} L_k(\theta_k,a_1) \tag{2}$$

$L_k(i,j)$ is the loss for k^{th} decision if $a_k = j$ and $\theta_k = i$. The optimal compound decision rule which minimizes the compound Bayes risk chooses $a_k\varepsilon A$, which minimizes

$$\sum_{\theta_k=1}^{r} L(\theta_k,a_k)p(\underset{\sim}{x}_n|\theta_n)G(\theta_k) \tag{3}$$

where $\underset{\sim}{x}_n$ is the set of recognition vectors for all cells of interest.

Note that $\underset{\sim}{x}_n$ is an n-dimensional vector of vectors. Further, $p(\underset{\sim}{x}_n|\theta_k)$ is a density function defined on this vector space which in practice can not be

evaluated without making some simplifying assumptions. We shall list the assumptions and state the final decision rule here without giving detailed derivations.

Consider Figure 1.

$$\begin{matrix} k_7 & k_2 & k_6 \\ k_3 & k & k_1 \\ k_8 & k_4 & k_5 \end{matrix}$$

Fig. 1. Labeling of neighbors of cell k.

Adjacent cells are defined as two cells with a common side. Our first assumption is that contextual relationships between nonadjacent cells are negligible. The second assumption is that the pattern vector x_k of a cell θ_k is a function only of θ_k, the nature of the cell.

The use of four neighbors (k_1, k_2, k_3, k_4) as contex information has been developed by Welch and Salter [6]. The final decision rule under the above assumptions is to choose $a \in A$ to minimize

$$\sum_{\theta_k=1}^{r} L(\theta_k,a)p(x_k|\theta_k)G(\theta_k) \prod_{i=1} \sum_{\theta_{k_i}} p(x_{k_i}|\theta_{k_i})p(\theta_{k_i}|\theta_k) \qquad (4)$$

We made one more assumption to incorporate eight neighbors of cell k as context in the algorithm for testing purposes. The additional assumption is that the contribution due to four adjacent cells of k is independent of that due to four cells, namely, k_5, k_6, k_7, k_8. The decision rule is then to choose $a \in A$ to minimize

$$\left[\sum_{\theta_k} L(\theta_k,a)p(x_k|\theta_k) \prod_{i=i}^{4} \sum_{\theta_{k_i}} p(x_{k_i}|\theta_{k_i})p(\theta_{k_i}|\theta_k)\right]_{adj} \times \left[p(x_k|\theta_k) \prod_{i=5}^{8} \sum_{k_i} p(x_{k_i}|\theta_{k_i})p(\theta_{k_i}|\theta_k) \cdot G(\theta_k)\right]_{con} \quad (5)$$

where the "adj" term denotes the contribution due to the four adjacent cells and the "con" term denotes that due to four corner cells.

Both of these context-dependent decision procedures have been programmed and applied to multispectral LANDSAT data. Initial results appear to verify that use of context can significantly improve classification accuracy. However, the cost in terms of computation required is substantial. In fact, the marginal improvements observed in going to 8 neighbors as compared to 4 may not be worth the added computational load. The potential impact of parallel processing on this problem also needs to be considered. It may also be possible to recast the equations in a recursive form which will reduce the required computations.

REFERENCES

[1] Fu, K.S., Syntactic methods in pattern recognition, Academic Press, 1974.

[2] Fu, K.S. and Swain, P.H., "On syntactic pattern recognition," Software Engineering, Vol. 2, J.T. Tou, ed., Academic Press, New York, 1971.

[3] Fu, K.S., "Pattern Recognition in Remote Sensing of the Earth's Resources," IEEE Transactions on Geoscience Electronics, Vol. GE-14, January 1976.

[4] Brayer, J.M. and Fu, K.S., "Web Grammars and Their Application to Pattern Recognition," Tech. Rept. TR-EE 75-1, Purdue University, December 1975.

[5] Kettig, R.L. and Landgrebe, D.A., "Classification of Multispectral Image Data by Extraction and Classification of Homogeneous Objects," Proc. Symp. on Machine Processing of Remotely Sensed Data, Purdue University, June 1975, IEEE Cat. No. 75CH1009-0C.

[6] Welch, J.R. and Salter, K.G., "A Context Algorithm for Pattern Recognition and Image Interpretation," IEEE Trans. Systems, Man and Cybernetics, Vol. SMC-1, No. 1, January 1971.

FILTERING TO REMOVE MULTIPLICATIVE NOISE IN COMPUTER SIMULATION OF CLOUDY PICTURES

O. R. Mitchell and P. L. Chen

I. INTRODUCTION

This report is a continuation of a previous report "Satellite Imagery Noise Removal" (ARPA Report, Nov. 1, 1974 to Oct. 31, 1975). We are using homomorphic filtering techniques to remove multiplicative noise effects such as those caused by cloud cover. The estimations of power spectrums of signal and noise have been revised. The filtering function is of the form $\frac{S_i}{S_i+N_i}$ where S_i is the power spectrum of the signal and N_i is of the noise. The detailed analysis of noise and homomorphic filtering process has been described in the previous report.

II. COMPUTER SIMULATION

A noisy picture was simulated using an original image $r(x,y)$ and a noise pattern $t(x,y)$ so that the output image $s(x,y)$ is formed by:

$$s(x,y) = I\, r(x,y)\, t(x,y) + I(1-t(x,y))$$

$$0 \leq r(x,y) \leq 1$$

$$0 \leq t(x,y) \leq 1$$

This represents the effect of cloud transmission $[t(x,y)]$, ground reflectance $[r(x,y)]$ and cloud reflection $[1-t(x,y)]$ of incident sunlight $[I]$.

Linear filtering is performed on

$$\log [I - s(x,y)] = \log [I - r(x,y)] + \log [t(x,y)] + \log I$$

which is additive signal and noise. The signal estimate is then obtained by exponentiating the filter output and inverting the grey levels.

III. SIMULATION RESULTS

Results of the simulation using 64x64 pictures are shown. For comparison purposes, the mean and standard deviation of the noisy and filtered pictures were normalized so that $\mu = 128$ and $\sigma = 48$ on a display scale of 0 to 255. Fig. 1(a) is a noisy signal with $I = 15$, $r(x,y) = 2/3$ in background and 4/5 in foreground, and $t(x,y)$ is white noise uniformly distributed between 0.4 and 0.6. The filtered result is shown in Fig. 1(b). Fig. 2(a) is another noisy picture with the noise increased to range from 0.02 to 1.0. Fig. 2(b) shows the filtered result. Fig. 3(a) is a noisy signal with $I = 15$, $r(x,y) = \frac{23}{30}$ in background and 4/5 in foreground, and the noise ranges uniformly from 0.4 to 0.6.

In Fig. 4(a) the same noise as used in Fig. 2(a) was low pass filtered before it was used. Fig. 4(b) shows the resulting filter output. The signal edges (high frequencies) are retained because the noise has no components at these frequencies.

Work is now in progress in applying a three dimensional extension of this filter to cloudy LANDSAT data.

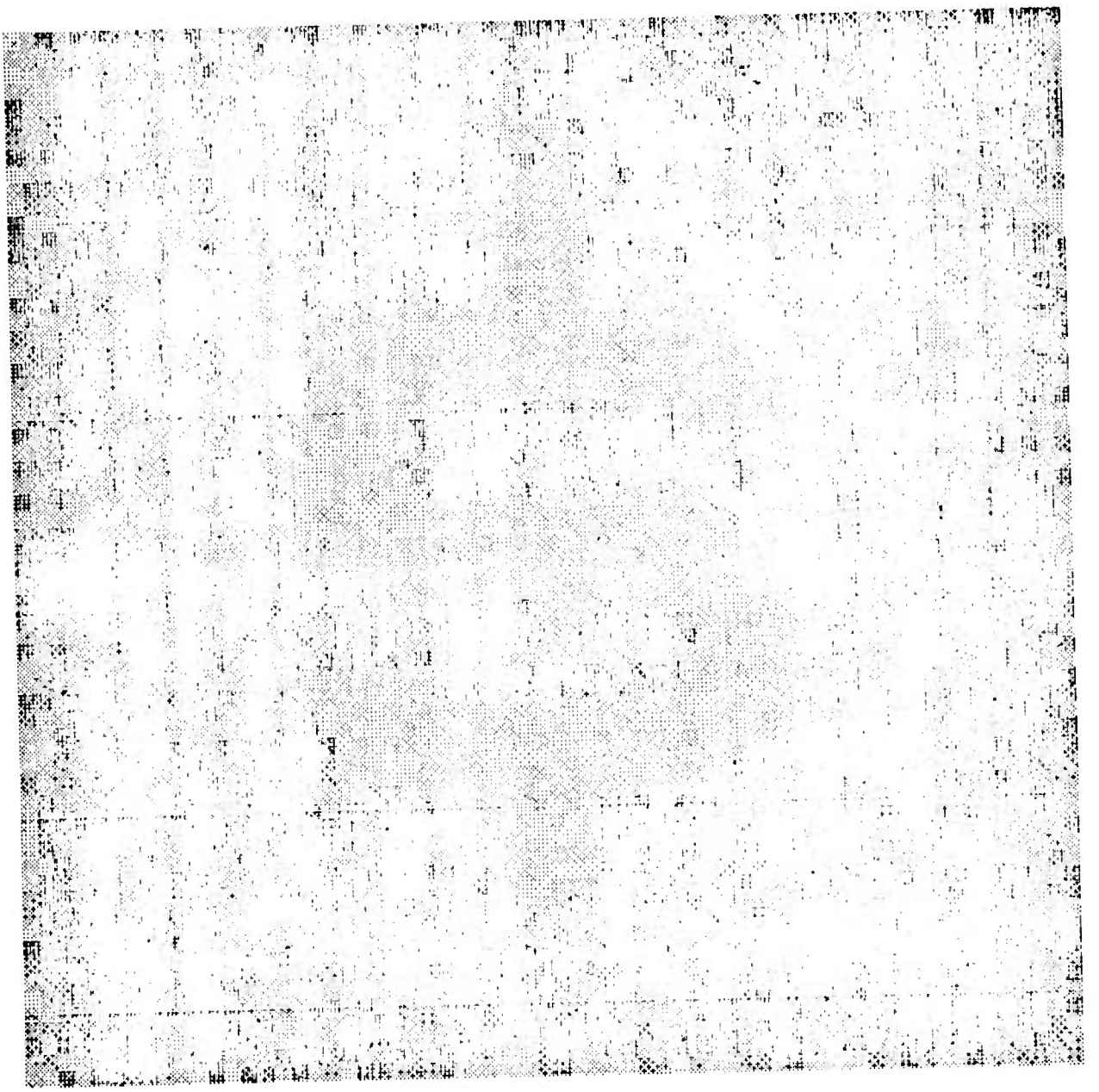


Fig. 1(a) Signal with multiplicative and reflective noise. Noise ranges from 0.4 to 0.6.

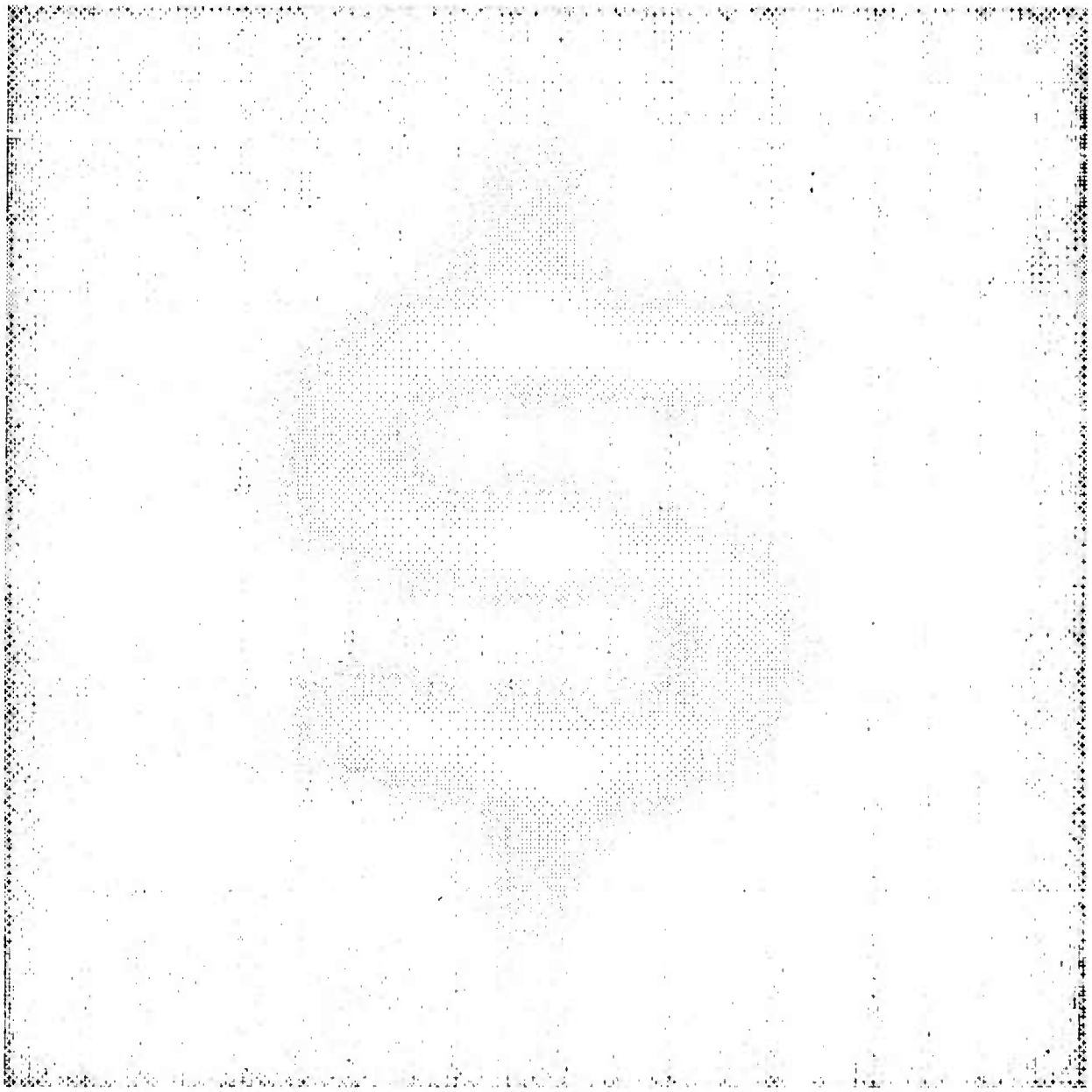


Fig. 1(b) Filtered output from Fig. 1(a), displayed with same mean and contrast as Fig. 1(a).

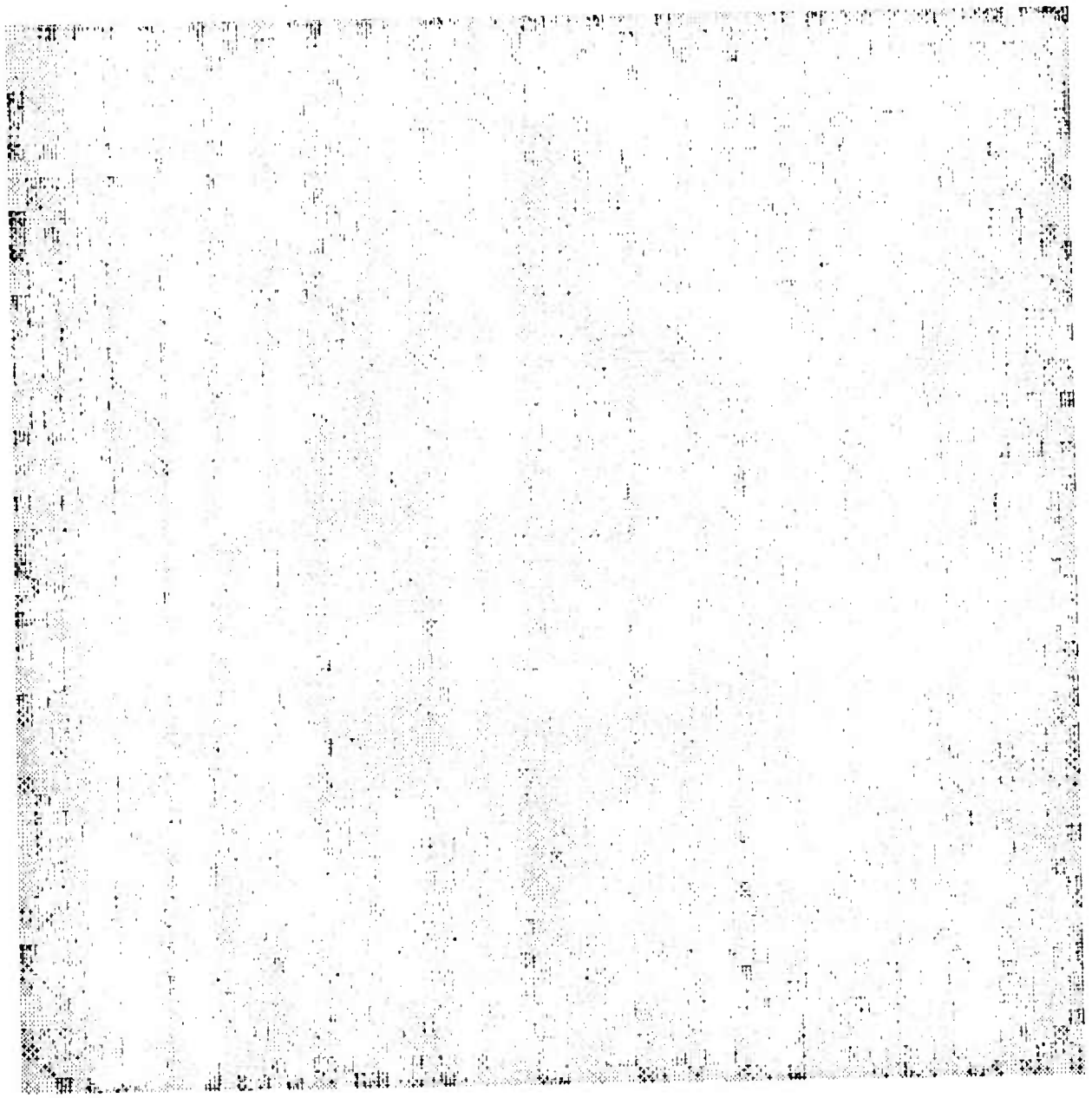


Fig. 2(a) Same picture as in Fig. 1(a) except the multiplicative noise is increased to 0.02 to 1.0.

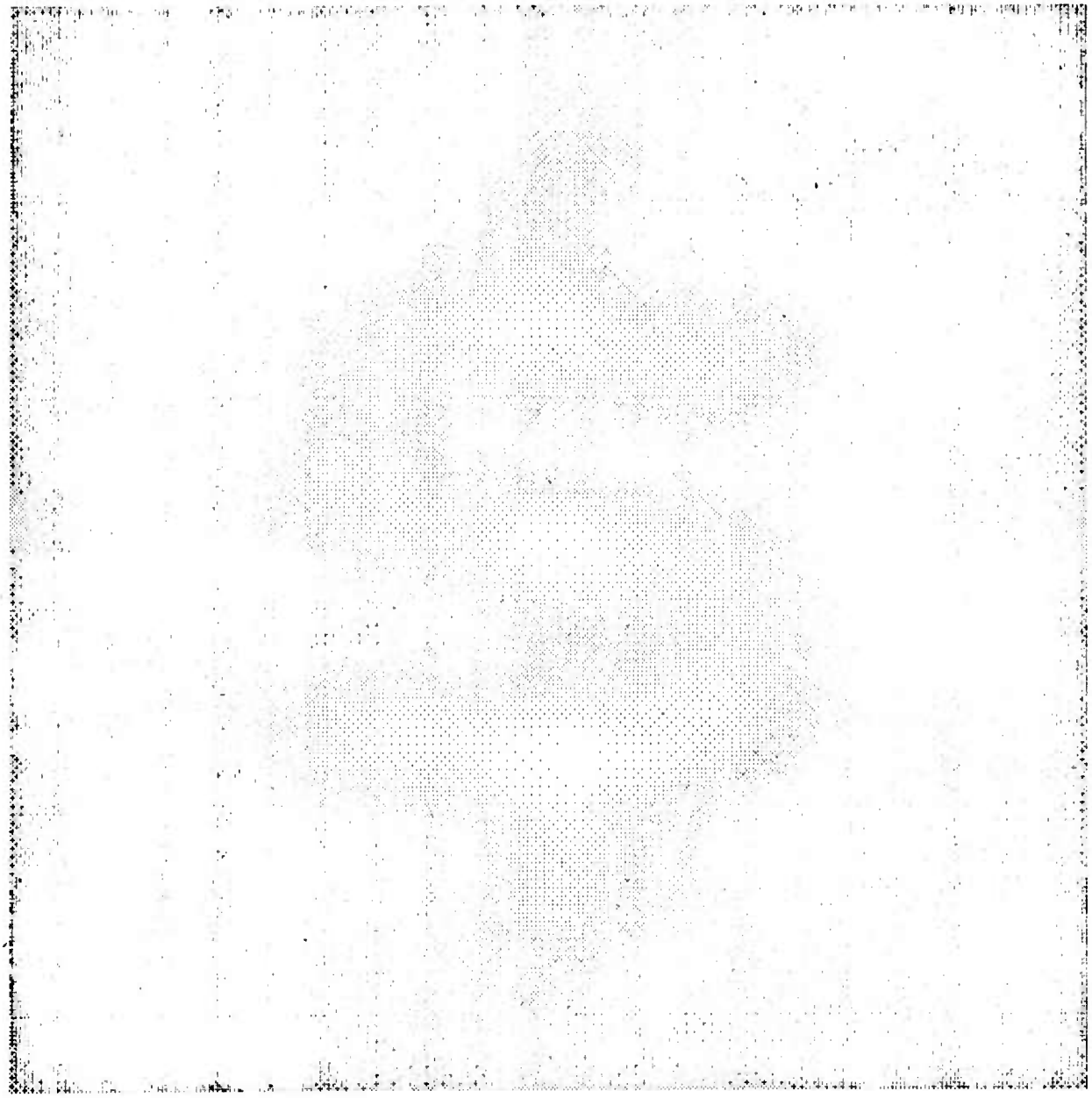


Fig. 2(b) Filtered output from Fig. 2(a)

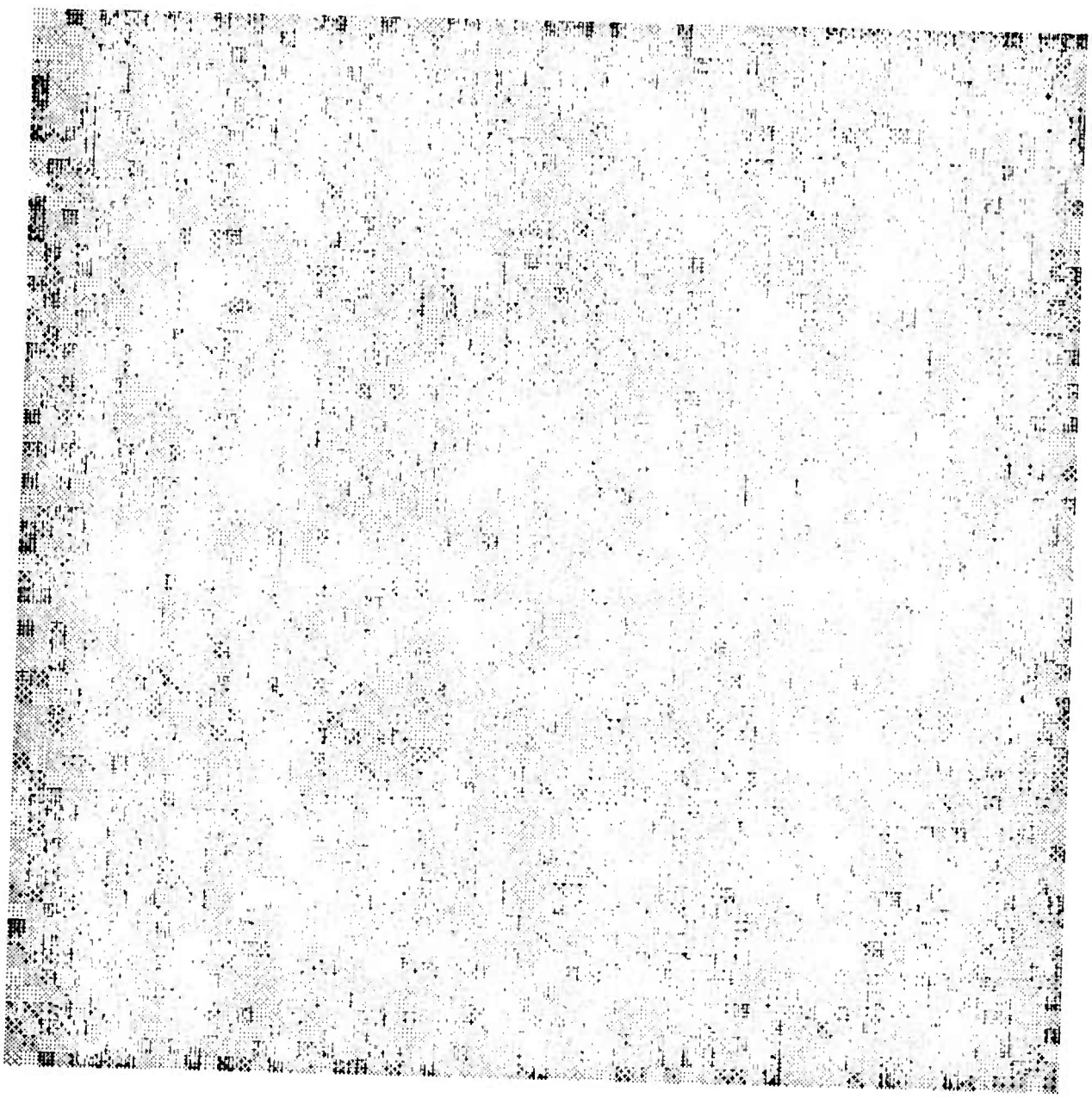


Fig. 3(a) Same noise as Fig. 1(a), signal level is further reduced.

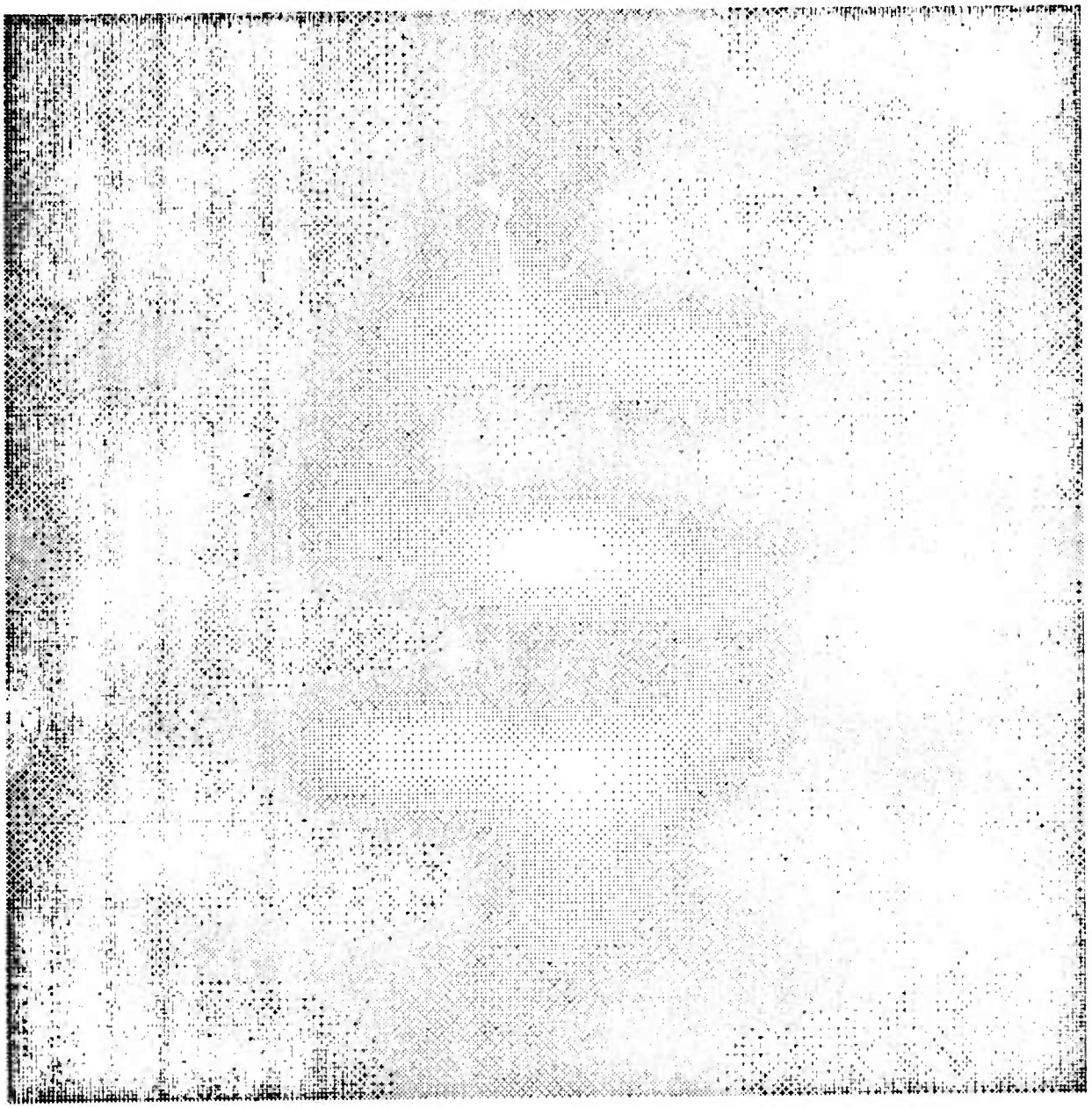


Fig. 3(b) Filtered output from Fig. 3(a).

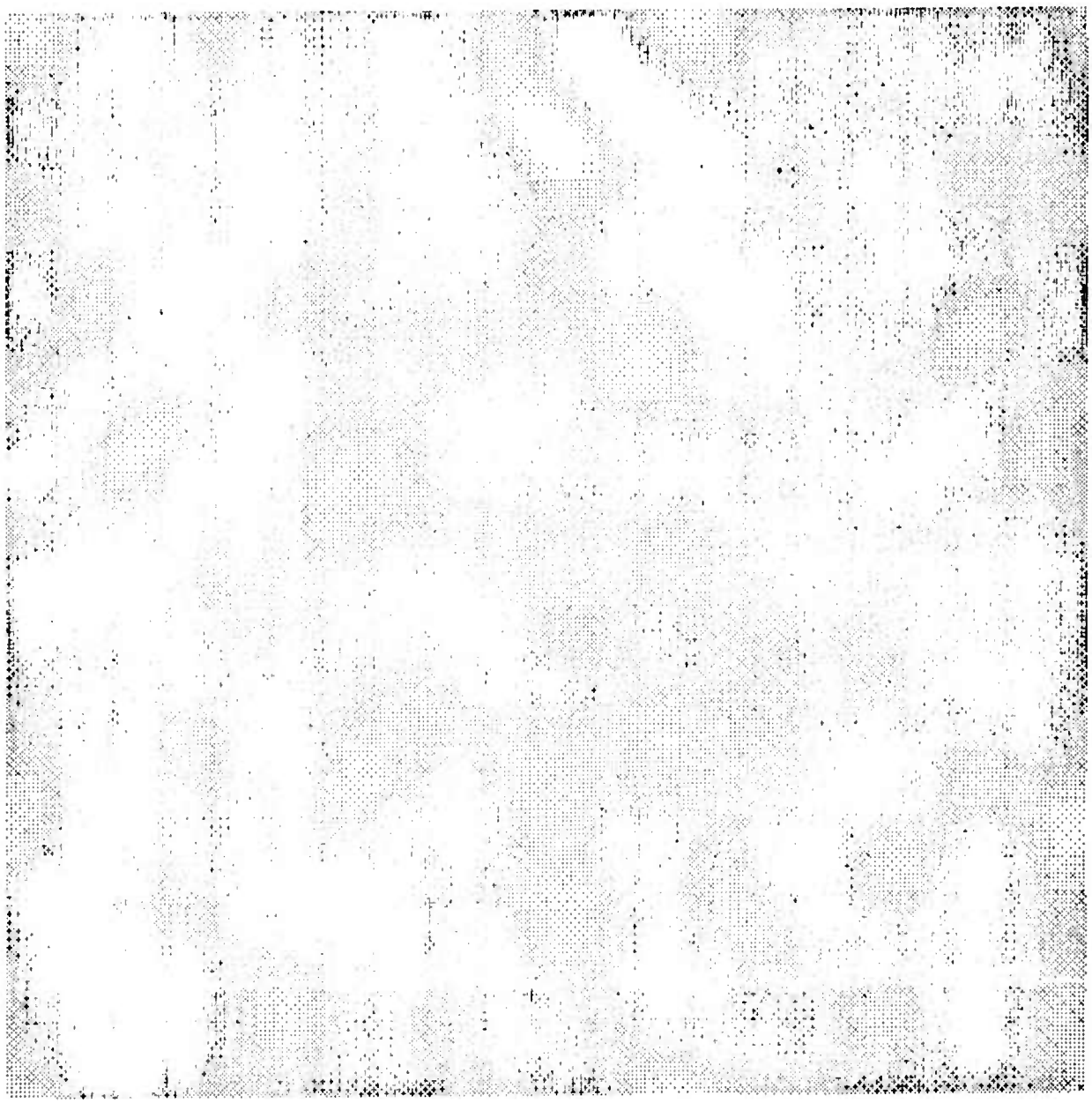


Fig. 4(a) Same picture as in Fig. 2(a) except the noise is first ideal low pass filtered to 8 cycles/picture width.

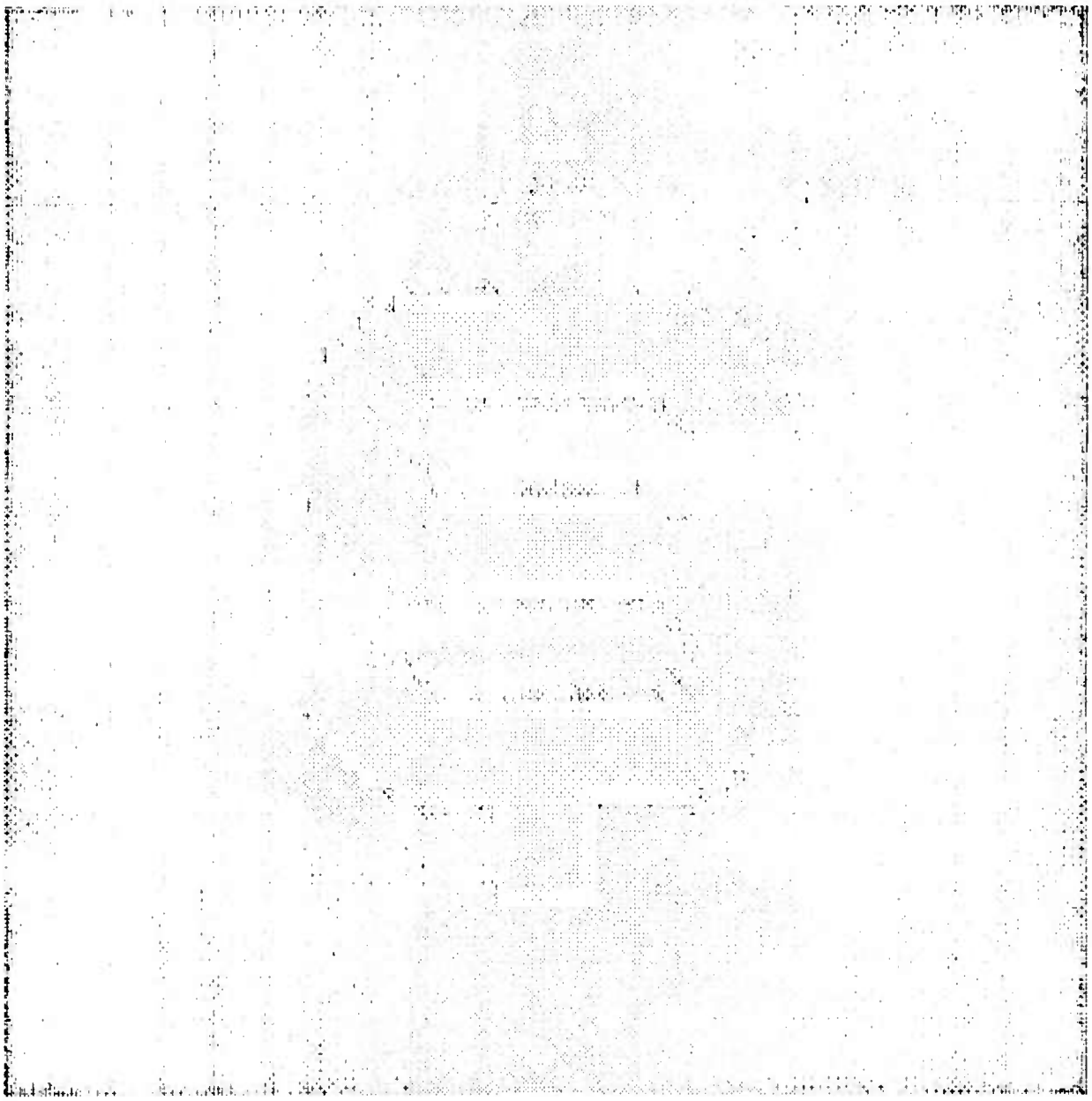


Fig. 4(b) Filtered output from Fig. 3(a).

MISSILE TRACKING

R. Gunshor and T.S. Huang

This project is motivated by the real-time video tracking problems at the U.S. Army White Sands Missile Range. The missile movie frames were supplied to us by Dr. Alton Gilbert of WSMR.

Our preliminary experimentation [1] has indicated that it is possible in many cases to detect and locate a missile by processing single scan lines. This opens up the possibility of very fast algorithms which scan selectively instead of full frame.

Solid state imagors, such as surface acoustical wave and charge-coupled devices, are particularly suitable for selective scanning. In this report, we present some very preliminary results of using a SAW linear scanner to scan one line of a missile frame. The purpose of the experiment is to test the capabilities of the SAW scanner under development at Purdue with regard to the missile tracking project.

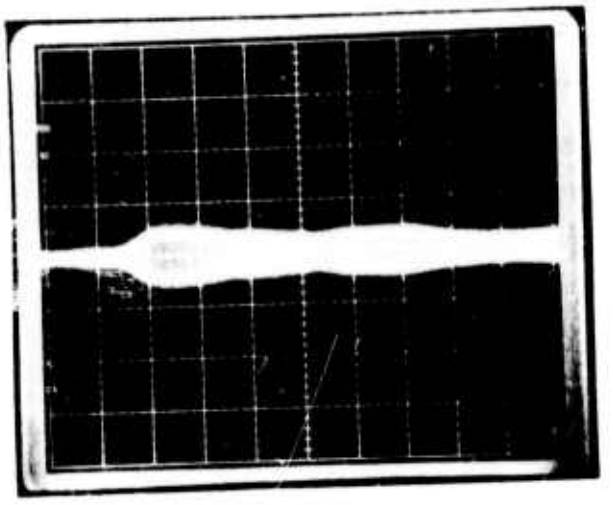
A frame from a motion picture of a missile flight at WSRM was scanned using a surface acoustic wave device. The results are shown in Fig. 1. Oscillograms A, B, and C represent the gate output at 2ω. The envelope amplitudes are proportional to the illumination levels along the scan path. A represents a scan of the background sky above the missile; B is a scan across the main missile body showing a bright area to the left, followed by a dark shadow region; C is a scan across the vapor trail of the missile showing the contrast between the vapor trail and the background sky.

The device employed in this experiment was a monolithic convolver fabricated by evaporation of CdSe onto a lithium niobate substrate. The convolver efficiency of this particular device was only -75 db (output divided by product of input powers). The state of the art for SAW convolvers is about -20 db.

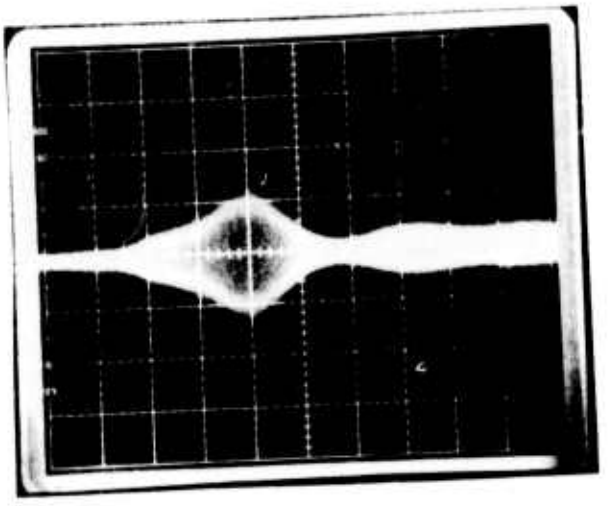
The low efficiency of this particular device meant that a relatively long scanning pulse (0.2 μsec) had to be used to obtain an acceptable signal-to-noise ratio. For a propagation velocity of 3.5×10^5 cm/s the spatial extent of this pulse is 0.7 mm; this is thus the order of resolution for this experiment. Also, the acoustic beam width is 2 mm; this results in some averaging along the length of the missile. Despite the use of this non-optimum scanning device, the main features of the picture are identifiable from the oscillograms. The output signals shown in the figure represent the rf output of the convolver gate. These signals (at 180 MHz for this device) can be applied directly to a SAW correlator for real-time processing of the image signals.

REFERENCE

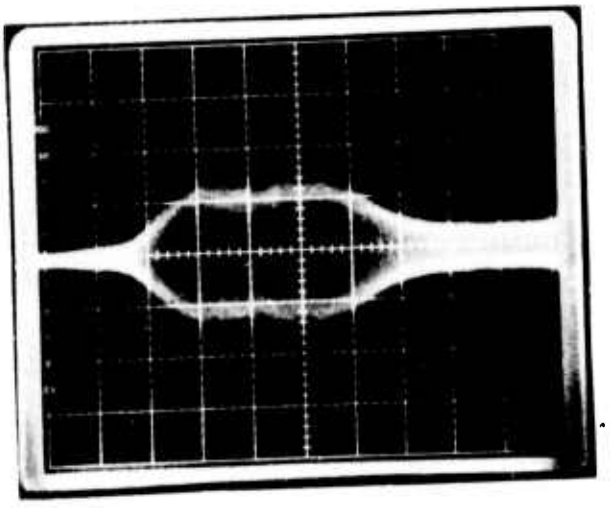
[1] T. S. Huang, and G. Y. Tang, "Missile tracking algorithms," in "Image Analysis and Modeling", Final Technical Report, March 1976, RADC-TR-76-76 (ARPA order 2893), Purdue University, (A023633).



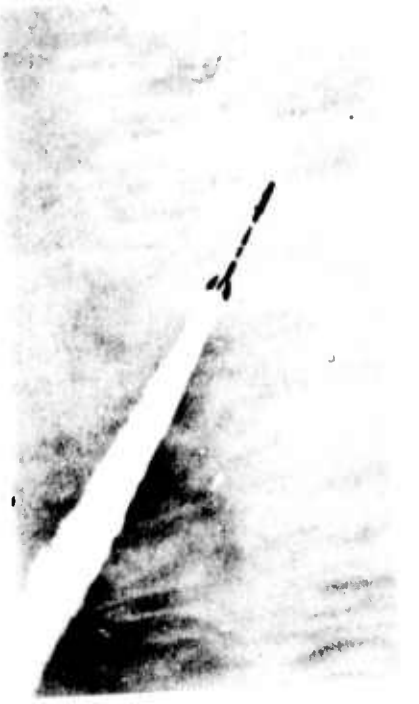
A



B



C



D

Fig. 1. Surface wave imaging experiment

FACILITIES

QTY	Manufacturer	Description
3	Beehive Elect.	"Super-Bee" Terminals
2	Tex. Inst.	"Silent 700" Terminals
1	Digi-Data	Industry standard magnetic tape system; 2, 9-track and 1, 7-track drives; one each NRZI and phase-encoded formatters/controllers
1	DEC	Dual-drive DECtape unit
1	DEC	RP03 disk drive (40 million characters)
1	Fabritek	96K-word auxiliary memory system (64K bought by ARPA, 32K by NASA)
1	Versatek	Electrostatic matrix printer
1	Comtal	Color picture display
1	Data Printer	132 column, 600 L.P.M. line printer
1	True-Data	Punched card reader
1	Tektronix	Model 4010, graphics display
1	DEC	PDP-11/45 computer system; system includes: 32K memory FPP-11 floating point processor (NSF money) H960 extension mounting cabinet 3 - small peripheral mountings blocks (DD-11) 1 UNIBUS repeater/expander DJ11, 16-line terminal multiplexor KW11-p programmed clock "ANTS" - type PDP-11/IMP interface

Note: Our PDP-11/45 is currently operating under the UNIX system.

BOOKS

FU, K.S., "ERTS Data Analysis," chapter in Applications of Syntactic Pattern Recognition, ed. , Springer-Verlag, 1976. (with J. Brayer, P. H. Swain)

HUANG, T., Image Processing and Digital Filtering (editor), Springer-Verlag, 1975.

SWAIN, P.H., "ERTS Data Analysis," chapter in Applications of Syntactic Pattern Recognition, K.S. Fu, ed., accepted for publication by Springer-Verlag, 1976.

WINTZ, P.A., "Picture Codng and Feature Extraction," chapter in Digital Image Processing, 1976.

JOURNAL PUBLICATIONS

FU, K.S., "Pattern Recognition in Remote Sensing of the Earth's Resources," IEEE Transactions on Geoscience Electronics, January 1976.

FU, K.S., "Error Estimation in Pattern Recognition via L^d-Distance Between Posteriori Density Functions," IEEE Transactions on Information Theory, Vol. IT-22, PP. 43-52, January 1976. (with T. Lissack)

FU, K.S., "A Tree System Approach for Fingerprint Pattern Recognition," IEEE Transactions on Computers, Vol. C-25, pp. 262-274, March 1976. (with B. Moayer)

FU, K.S., "An Application of Stochastic Languages to Fingerprint Pattern Recognition," Pattern Recognition, Vol. 8, pp. 175-181, 1976. (with B. Moayer)

FU, K.S., "A Minicomputer Facility for Picture Processing and Pattern Recognition Research," COMPUTER, Vol. 9, pp. 70-77, May 1976. (with E. Persoon)

FU, K. S., "Parametric Feature Extraction Through Error Minimization, Applied to Medical Diagnosis," IEEE Transactions on Systems, Man and Cybernetics, Vol. SMC-6, September 1976. (with T. Lissack)

HUANG, T., "The Importance of Phase in Image Processing Filters," IEEE Trans. on Acoustics, Speech, and Signal Processing, December 1975. (with J. Burnett and A. Deczky)

HUANG, T., "Facsimile Coding by Skipping White," IEEE Trans. on Comm., December 1975. (with A. Hussian)

MITCHELL, O.R., "Effect of Spatial Frequency on the Visibility of Unstructed Patterns," J. Opt. Soc. Am., Vol. 66, No. 4, April 1976.

MITCHELL, O.R., "Digital Communications Equipment for Instructional Purposes," IEEE Trans. on Education, May 1976. (with W. L. Thomas)

SWAIN, P.H., "Determining Density of Maize Canopy from Digitized Photographic Data," Agronomy Journal, Vol. 68, pp. 55-59, January-February 1976. (with E. R. Stoner and M. F. Baumgardner)

SWAIN, P.H., "Selective Radiant Temperature Mapping Using a Layered Classifier," IEEE Trans. Geoscience Electronics, Vol. GE-14, pp. 101-106, April 1976. (with L. A. Bartolucci and C. L. Wu)

CONFERENCES

FU, K.S., "Processing of Chest X-Ray Images by Computer," IFIP Working Conference on Decision-Making and Medical Care, May 24-29, 1976, Dijon, France.

FU, K.S., "A Syntactic Approach to the Representation of Image Structures," IFIP Working Conference on Environmental Modelling, April 26-28, 1976, Tokyo, Japan.

FU, K.S., "High Dimensional Languages and Grammatical Inference," IEEE Joint Workshop on Pattern Recognition and Artificial Intelligence, June 1-3, 1976, Hyannis, MA.

FU, K.S., "Some Applications of Stochastic Languages," Symposium on Application of Statistics, June 14-18, 1976, Dayton, Ohio.

FU, K.S., "Tree System Approach for LANDSAT Data Interpretation," Proc. Symp. on Machine Processing of Remotely Sensed Data, June 29-July 1, 1976.

FU, K.S., An Approach to the Design of a Linear Binary Tree Classifier," Proc. Symp. on Machine Processing of Remotely Sensed Data, June 29-July 1, 1976.

FU, K.S., "The Linguistic Approach to Pattern Recognition," Advanced Seminar on Classification and Clustering, The Mathematics Research Center, University of Wisconsin, Madison, Wisconsin, May 3-5, 1976.

HUANG, T., "Facsimile Coding by Skipping White," presented at the 1976 Picture Coding Symposium, Asilomar, January 28-30, 1976.

HUANG, T., "Restoration of Images Degraded by Spatially-Varying Systems," presented at the OSA Technical Meeting on Image Processing, Asilomar, February 24-25, 1976.

HUANG, T., "Nonlinear Estimation of Markov Jump Processes," to be presented at the IEEE Int'l Information Theory Symposium, Ronneby, Sweden, June 21-24, 1976. (with J. Burnett)

HUANG, T., "Digital Straight Edges," presented at the 6th Annual Symp. on Automatic Imagery Pattern Recognition, Univ. of Maryland, Silver Spring, MD, June 1-2, 1976. (with G. Tang)

HUANG, T., "Two-Dimensional Fourier Transforms," "Image Restoration," and "Film Models," presented at NATO Advanced Institute on Digital Image Processing and Analysis, Bonas, France, June 14-25, 1976.

HUANG, T., "Image Processing Research at Purdue University," USSR-US joint workshop on Inf. Theo., December 14-19, 1975, Moscow, U.S.S.R.

MITCHELL, O.R., "Texture Edge Detection and Classification Using Max-Min Descriptors," Sixth Annual Symposium on Automatic Imagery Pattern Recognition, Univ. of Maryland, College Park, MD, June 1-2, 1976.

MITCHELL, O.R., "Filtering to Remove Cloud Cover in Satellite Imagery," LARS Symposium, Machine Processing of Remotely Sensed Data, June 29-July 1, 1976, West Lafayette, IN. (with P. L. Chen)

SWAIN, P.H., "Application of a Class of Sequential Classifiers to Multitemporal Remote Sensing Data," Proc. Third Symposium on Machine Processing of Remotely Sensed Data, June 1976. (with H. Hauska)

SWAIN, P.H., "Some Time for Texture in the Spectrum of Spatial Features," presented at the Engineering Foundation Conference on Algorithms for Image Processing, Franklin Pierce College, Rindge, New Hampshire, August 1976. (with D. A. Landgrebe)

WINTZ, P.A., "Digital Telephony and Signal Processing Theory and Practice," Instanbul, Turkey, December 1975. (with J. Sergo)

WINTZ, P.A., "Images and Models for Image Noise," presented at NATO Advanced Institute on Digital Image Processing and Analysis, Bonas, France, June 14-25, 1976.

WINTZ, P.A., "Image Coding with Emphasis on Techniques for Producing Decorrelated Image Data," presented at NATO Advanced Institute on Digital Image Processing and Analysis, Bonas, France, June 14-25, 1976.

STAFF

CO-PRINCIPAL INVESTIGATORS

T. S. Huang
K. S. Fu

PROFESSORIAL

K. Fukunaga
O. Mitchell
P. Swain
P. Wintz

GRADUATE RESEARCHERS

S. Berger
J. Burnett
Wm. Chan
P.H. Chen
P.L. Chen
X. Dang
R. Florek
J. Keng
R.L. Li
P. Narendra
B. O'Connor
D. Panda
A. Salahi
G. Tang
T. Wallace
M. Yoo
T.S. Yu

RESEARCH STAFF

J. Besemer
W. Robey

UNDERGRADUATE RESEARCHERS

C. Buckwacter
M. DeMoney
R. Johnson
J. Schwab

ELECTRONIC TECHNICIANS

D. Azpell

SECRETARIES

M. Barbour
M. Claire

*U.S. GOVERNMENT PRINTING OFFICE:1976-714-025/22

MISSION
of
Rome Air Development Center

RADC plans and conducts research, exploratory and advanced development programs in command, control, and communications (C^3) activities, and in the C^3 areas of information sciences and intelligence. The principal technical mission areas are communications, electromagnetic guidance and control, surveillance of ground and aerospace objects, intelligence data collection and handling, information system technology, ionospheric propagation, solid state sciences, microwave physics and electronic reliability, maintainability and compatibility.